

[illegible][illegible]

```
DDDDDDDD  RRRRRRRR  DDDDDDDD  RRRRRRRR  IIIIII  VV  VV  EEEEEEEEE  RRRRRRRR
DDDDDDDD  RRRRRRRR  DDDDDDDD  RRRRRRRR  IIIIII  VV  VV  EEEEEEEEE  RRRRRRRR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RRRRRRRR  DD  DD  RRRRRRRR  II  VV  VV  EE  RRRRRRRR
DD  DD  RRRRRRRR  DD  DD  RRRRRRRR  II  VV  VV  EEEEEEEE  RRRRRRRR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DD  DD  RR  RR  DD  DD  RR  RR  II  VV  VV  EE  RR  RR
DDDDDDDD  RR  RR  DDDDDDDD  RR  RR  IIIIII  VV  VV  EEEEEEEEE  RR  RR
DDDDDDDD  RR  RR  DDDDDDDD  RR  RR  IIIIII  VV  VV  EEEEEEEEE  RR  RR
                                     ....
```

```
LL  IIIIII  SSSSSSSS
LL  IIIIII  SSSSSSSS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SSSSSS
LL  II  SSSSSS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```



DRDRIVER  
Table of contents

- RM03/RM05/RM80/RP07 DISK DRIVER J 16

15-SEP-1984 23:52:45 VAX/VMS Macro V04-00

Page 0

(1)	447	FUNCTION DECISION TABLE
(1)	562	START I/O OPERATION
(1)	1102	HARDWARE FUNCTION EXECUTION
(1)	1606	REGISTER DUMP ROUTINE
(1)	1647	DISK DRIVE INITIALIZATION
(1)	1783	UNSOLICITED INTERRUPT ROUTINE
(1)	1818	CLASSIFY DRIVE TYPE AND SET PARAMETERS



```
0000 1 .TITLE DRDRIVER - RM03/RM05/RM80/RP07 DISK DRIVER
0000 2 .IDENT 'V04-001'
0000 3
0000 4
0000 5 *****
0000 6
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28 D. N. CUTLER, LEN KAWELL 23-NOV-77
0000 29
0000 30 MODIFIED BY:
0000 31
0000 32 V04-001 PRD0112 Paul R. DeStefano 06-Sep-1984
0000 33 Modify ECC routine to allow for RP07's handling of
0000 34 HCRC errors as class A errors when HCI is set.
0000 35
0000 36 Add sanity check to offset recovery routine to insure
0000 37 that there is data to be transferred before offset
0000 38 recovery is performed.
0000 39
0000 40 V03-016 RAS0300 Ron Schaefer 27-Apr-1984
0000 41 Add DEV$M_NNM characteristic to DECHAR2 so that these
0000 42 devices will have the 'node$' prefix.
0000 43
0000 44 V03-015 PRD0081 Paul R. DeStefano 19-Mar-1984
0000 45 For dual ported drives, make sure the port isn't
0000 46 reseized by the time we come off the I/O fork queue.
0000 47
0000 48 V03-014 PRD0048 Paul R. DeStefano 01-Feb-1984
0000 49 Fix context used in TIMEWAIT macro when referencing
0000 50 device registers.
0000 51
0000 52 V03-013 PRD0036 Paul R. DeStefano 09-Sep-1983
0000 53 Added EXES$LCLDSKVALID to function decision table.
0000 54
0000 55 V03-012 ROW0211 Ralph O. Weber 16-AUG-1983
0000 56 Change device-dependent UCB definition base from UCB$W_BCR+2
0000 57 to UCB$K_LCL_DISK_LENGTH.
```



```
0000 58 :
0000 59 :
0000 60 :
0000 61 :
0000 62 :
0000 63 :
0000 64 :
0000 65 :
0000 66 :
0000 67 :
0000 68 :
0000 69 :
0000 70 :
0000 71 :
0000 72 :
0000 73 :
0000 74 :
0000 75 :
0000 76 :
0000 77 :
0000 78 :
0000 79 :
0000 80 :
0000 81 :
0000 82 :
0000 83 :
0000 84 :
0000 85 :
0000 86 :
0000 87 :
0000 88 :
0000 89 :
0000 90 :
0000 91 :
0000 92 :
0000 93 :
0000 94 :
0000 95 :
0000 96 :
0000 97 :
0000 98 :
0000 99 :
0000 100 :
0000 101 :
0000 102 :
0000 103 :
0000 104 :
0000 105 :
0000 106 :
0000 107 :
0000 108 :
0000 109 :
0000 110 :
0000 111 :
0000 112 :
0000 113 :
0000 114 :
```

V03-011 WMC0001 Wayne Cardoza 09-Aug-1983  
Missing G^.

V03-010 KDM0060 Kathleen D. Morse 14-Jul-1983  
Replace reference to IPR TODR with call to cpu-dependent  
routine, EXESREAD\_TODR.  
Add \$DEVDEF.

V03-009 PRD0027 Paul R. DeStefano 17-Jun-1983  
Modified EXFNC routine to bypass setting of offset mode  
for RP07's to prevent RP07 microcode hang and system crash.

V03-008 PRD0023 Paul R. DeStefano 05-May-1983  
Modified ERROR routine to attempt to clear a drive  
unsafe condidtion.

V03-007 PRD53302 Paul R. DeStefano 05-May-1983  
ECO 02 Modified RETRYERR routine to issue a Drive Clear before  
retrying a function. Modified FUNCXT routine to issue  
a Drive Clear function before releasing the drive.

V03-006 PRD0018 Paul R. DeStefano 26-Apr-1983  
Modified FATALERR routine to return \$\$\$\_PARITY only for  
errors that possibly indicate bad media. All other error  
conditions which formerly returned \$\$\$\_PARITY now return  
\$\$\$\_CNTLERR.

V03-005 PRD0015 Paul R. DeStefano 26-Apr-1983  
Modified ECC correction logic so that ECC is only applied  
when there is single bit ECC correctable error, or if there  
is a multiple bit ECC correctable error and the error cannot  
be corrected using retries.

V03-004 ROW47161 Ralph O. Weber 16-SEP-1982  
ECO 01 Enhance ECC recovery logic to prevent bytes transfered counts  
which are not exact multiples of 512 from causing transfer  
parameters from being incorrectly updated. Because a non-512-  
intergal bytes transfered counts indicates an incomplete  
transfer of the last block, this change also prevents ECC  
corrections when such bytes transfered counts are encountered.

V03-003 KDM0002 Kathleen D. Morse 28-Jun-1982  
Added \$DCDEF, \$DYNDEF, and \$\$\$DEF.

V03-002 KTA0100 Kerbey T. Altmann 07-Jun-1982  
Add code to set UCBSL\_MEDIA\_ID.

RM03/RM05/RM80/RP07 DISK DRIVER

MACRO LIBRARY CALLS

\$CRBDEF :DEFINE CRB OFFSETS  
\$DCDEF :DEFINE DEVICE CLASSES  
\$DDBDEF :DEFINE DDB OFFSETS



```
0000 115      $DEVDEF      ;DEFINE DEVICE CHARACTERISTICS
0000 116      $DPTDEF      ;DEFINE DPT OFFSETS
0000 117      $DYNDEF      ;DEFINE DYNAMIC DATA STRUCTURE TYPES
0000 118      $EMBDEF      ;DEFINE EMB OFFSETS
0000 119      $IDBDEF      ;DEFINE IDB OFFSETS
0000 120      $IODEF       ;DEFINE I/O FUNCTION CODES
0000 121      $IRPDEF      ;DEFINE IRP OFFSETS
0000 122      $MBADEF      ;DEFINE MBA REGISTER OFFSETS
0000 123      $PRDEF       ;DEFINE PROCESSOR REGISTER NUMBERS
0000 124      $SSDEF       ;DEFINE SYSTEM STATUS CODES
0000 125      $UCBDEF      ;DEFINE UCB OFFSETS
0000 126      $VECDDEF     ;DEFINE INTERRUPT DISPATCH VECTOR OFFSETS
0000 127
0000 128      :
0000 129      : LOCAL MACROS
0000 130      :
0000 131      : EXECUTE FUNCTION AND BRANCH ON RETRIABLE ERROR CONDITION
0000 132      :
0000 133      :
0000 134      .MACRO EXFUNC BDST,FCODE
0000 135      .IF NB FCODE
0000 136      MOVZBL #CD'FCODE,R0
0000 137      .ENDC
0000 138      BSBW FEX
0000 139      .SIGNED_WORD BDST--2
0000 140      .ENDM
0000 141
0000 142      :
0000 143      : GENERATE FUNCTION TABLE ENTRY AND CASE TABLE INDEX SYMBOL
0000 144      :
0000 145      :
0000 146      .MACRO GENF FCODE
0000 147      CD'FCODE=-FTAB
0000 148      .BYTE FCODE!RM_CS1_M_GO
0000 149      .ENDM
0000 150
0000 151      :
0000 152      : LOCAL SYMBOLS
0000 153      :
0000 154      : MASSBUS REGISTER OFFSETS
0000 155      :
0000 156      :
0000 157      $DEFINI RM
0000 158
0000 159 $DEF RM_CS1 .BLKL 1 ;DRIVE CONTROL REGISTER
0004 160 _VIELD RM_CS1,0,<- ;DRIVE CONTROL REGISTER BIT DEFINITIONS
0004 161 <GO,,M>,- ;GO BIT
0004 162 <FCODE,5>- ;FUNCTION CODE
0004 163 >
0004 164 $DEF RM_DS .BLKL 1 ;DRIVE STATUS REGISTER
0008 165 _VIELD RM_DS,0,<- ;DRIVE STATUS REGISTER BIT DEFINITIONS
0008 166 <OM,,M>,- ;OFFSET MODE
0008 167 <,5>- ;RESERVED BITS
0008 168 <VV,,M>,- ;VOLUME VALID
0008 169 <DRY,,M>,- ;DRIVE READY
0008 170 <DPR,,M>,- ;DRIVE PRESENT
0008 171 <PGM,,M>- ;PROGRAMMABLE
```



```
0008 172 <LST,,M>,- : LAST SECTOR TRANSFERED
0008 173 <WRL,,M>,- : DRIVE WRITE LOCKED
0008 174 <MOL,,M>,- : MEDIUM ONLINE
0008 175 <PIP,,M>,- : POSITIONING IN PROGRESS
0008 176 <ERR,,M>,- : COMPOSITE ERROR
0008 177 <ATA,,M>,- : ATTENTION ACTIVE
0008 178 >
0008 179 $DEF RM_ER1 .BLKL 1 : ERROR REGISTER 1
000C 180 _VIELD RM_ER1,0,<- : ERROR REGISTER 1 BIT DEFINITIONS
000C 181 <ICF,,M>,- : ILLEGAL FUNCTION
000C 182 <ILR,,M>,- : ILLEGAL REGISTER
000C 183 <RMR,,M>,- : REGISTER MODIFY REFUSED
000C 184 <PAR,,M>,- : PARITY ERROR
000C 185 <FER,,M>,- : FORMAT ERROR
000C 186 <WCF,,M>,- : WRITE CLOCK FAIL
000C 187 <ECH,,M>,- : ECC HARD ERROR
000C 188 <HCE,,M>,- : HEADER COMPARE ERROR
000C 189 <HCRC,,M>,- : HEADER CRC ERROR
000C 190 <AOE,,M>,- : ADDRESS OVERFLOW ERROR
000C 191 <IAE,,M>,- : ILLEGAL ADDRESS ERROR
000C 192 <WLE,,M>,- : WRITE LOCK ERROR
000C 193 <DTE,,M>,- : DRIVE TIMING ERROR
000C 194 <OPI,,M>,- : OPERATION INCOMPLETE
000C 195 <UNS,,M>,- : DRIVE UNSAFE
000C 196 <DCK,,M>,- : DATA CHECK ERROR
000C 197 >
000C 198 $DEF RM_MR .BLKL 1 : MAINTENANCE REGISTER
0010 199 _VIELD RM_MR,0,<- : MAINTENANCE REGISTER DEFINITIONS
0010 200 <PAR,8>,- : DIAGNOSTIC PARAMETER
0010 201 <RTN,7>,- : DIAGNOSTIC ROUTINE NUMBER
0010 202 <DM,,M>,- : DIAGNOSTIC MODE
0010 203 >
0010 204 $DEF RM_AS .BLKL 1 : ATTENTION SUMMARY REGISTER
0014 205 $DEF RM_DA .BLKL 1 : DESIRED SECTOR/TRACK ADDRESS REGISTER
0018 206 _VIELD RM_DA,0,<- : DESIRED ADDRESS FIELD DEFINITIONS
0018 207 <SA,5>,- : DESIRED SECTOR ADDRESS
0018 208 <3>,- : RESERVED BITS
0018 209 <TA,5>,- : DESIRED TRACK ADDRESS
0018 210 >
0018 211 $DEF RM_DT .BLKL 1 : DRIVE TYPE REGISTER
001C 212 _VIELD RM_DT,0,<- : DRIVE TYPE REGISTER FIELD DEFINITIONS
001C 213 <DTN,9>,- : DRIVE TYPE NUMBER
001C 214 <2>,- : RESERVED BITS
001C 215 <DRQ,,M>,- : DRIVE REQUEST REQUIRED
001C 216 >
001C 217 $DEF RM_LA .BLKL 1 : LOOKAHEAD REGISTER
0020 218 $DEF RM_SN .BLKL 1 : SERIAL NUMBER REGISTER
0024 219 $DEF RM_OF .BLKL 1 : OFFSET REGISTER
0028 220 _VIELD RM_OF,0,<- : OFFSET REGISTER BIT DEFINITIONS
0028 221 <OFF,8>,- : OFFSET VALUE
0028 222 <1>,- : RESERVED
0028 223 <SSEI,,M>,- : SKIP SECTOR INHIBIT (RM80)
0028 224 <HCI,,M>,- : HEADER COMPARE INHIBIT
0028 225 <ECI,,M>,- : ECC INHIBIT (avoid using this bit)
0028 226 <FMT,,M>,- : 16-BIT FORMAT
0028 227 <1>,- : RESERVED
0028 228 <MTD,,M>,- : MOVE TRACK DESCRIPTOR
```



```
0028 229 <CMO,,M>- ; COMMAND MODIFIER
0028 230 > ;
0028 231 $DEF RM_DC .BLKL 1 ; DESIRED CYLINDER ADDRESS
002C 232 $DEF RM_UNUSED .BLKL 1 ; UNUSED
0030 233 $DEF RM_MR2 .BLKL 1 ; MAINTENANCE REGISTER 2
0034 234 $DEF RM_ER2 .BLKL 1 ; ERROR REGISTER 2
0038 235 -VIELD RM_ER2,3,<- ; ERROR REGISTER 2 BIT DEFINITIONS
0038 236 <DPE,,M>- ; DATA PARITY ERROR
0038 237 <,1>- ; RESERVED BIT
0038 238 <SSE,,M>- ; SKIP SECTOR ERROR (RM80)
0038 239 <,1>- ; RESERVED BIT
0038 240 <DVC,,M>- ; DEVICE CHECK ERROR
0038 241 <,2>- ; RESERVED BITS
0038 242 <LBC,,M>- ; LOSS OF BIT CLOCK ERROR
0038 243 <LSC,,M>- ; LOSS OF SYSTEM CLOCK ERROR
0038 244 <IVC,,M>- ; INVALID COMMAND ERROR
0038 245 <OPE,,M>- ; OPERATOR PLUG ERROR
0038 246 <SKI,,M>- ; SEEK INCOMPLETE ERROR
0038 247 <BSE,,M>- ; BAD SECTOR ERROR
0038 248 > ;
0038 249 $DEF RM_EC1 .BLKL 1 ; ECC POSITION REGISTER
003C 250 VIELD RM_EC1,0,<<POS,13>> ; ECC POSITION FIELD
003C 251 $DEF RM_EC2 .BLKL 1 ; ECC PATTERN REGISTER
0040 252 -VIELD RM_EC2,0,<<PAT,11>> ; ECC PATTERN FIELD
0040 253
0040 254 $DEFEND RM
0000 255
0000 256 ;
0000 257 ; DEFINE DEVICE DEPENDENT UNIT CONTROL BLOCK OFFSETS
0000 258 ;
0000 259
0000 260 $DEFINI UCB
0000 261
000000CC 0000 262 .=UCBSK_LCL_DISK_LENGTH ; Establish device-dependent UCB base
00CC 263
00CC 264 $DEF UCBSL_DR_SR .BLKL 1 ; SAVED MBA STATUS REGISTER
00D0 265 $DEF UCBSW_DR_ER2 .BLKW 1 ; SAVED ERROR REGISTER 2
00D2 266 $DEF UCBSW_DR_MR .BLKW 1 ; MAINTENANCE REGISTER
00D4 267 $DEF UCBSB_DR_SSTS .BLKB 1 ; SOFTWARE STATUS BYTE
00D5 268 -VIELD DR,0,<- ; SOFTWARE STATUS BIT DEFINITIONS
00D5 269 <DCK,,M>- ; DATACHECK IN PROGRESS
00D5 270 <OM,,M>- ; OFFSET MODE
00D5 271 <NOECC,,M>- ; Don't correct with ECC
00D5 272 <DUALPORT,,M>- ; Drive has a dualport kit
00D5 273 <ECC_DEFER,,M>- ; Flag to indicate that ECC correction
00D5 274 > ; has been deferred until offset
00D5 275 ; retries are exhausted.
00D5 276 $DEF UCBSB_DR_ERL .BLKB 1 ; ERROR LOGGING REGISTER FOR MED OFFLINE
00D6 277 $DEF UCBSW_DR_OFRR .BLKW 1 ; SAVED OFFSET REGISTER
00D8 278 $DEF UCBSL_DR_BCR .BLKL 1 ; Saved (longword) MBA byte count reg.
000000DC 00DC 279 UCBSK_DR_LENGTH=.
00DC 280
00DC 281 $DEFEND UCB
0000 282
0000 283 ;
0000 284 ; HARDWARE FUNCTION CODES
0000 285 ;
```



```
00000000 0000 286
00000004 0000 287 F_NOP=0*2 ;NO OPERATION
00000006 0000 288 F_SEEK=2*2 ;SEEK CYLINDER
00000008 0000 289 F_RECAL=3*2 ;RECALIBRATE
0000000A 0000 290 F_DRVCLR=4*2 ;DRIVE CLEAR
0000000C 0000 291 F_RELEASE=5*2 ;RELEASE DRIVE
0000000E 0000 292 F_OFFSET=6*2 ;OFFSET HEADS
00000010 0000 293 F_RETCENTER=7*2 ;RETURN TO CENTERLINE
00000012 0000 294 F_READPRESET=8*2 ;READ IN PRESET
00000014 0000 295 F_PACKACK=9*2 ;PACK ACKNOWLEDGE
00000016 0000 296 F_SEARCH=12*2 ;SEARCH FOR SECTOR
00000018 0000 297 F_SEARCHA=12*2 ;SEARCH AHEAD FOR SECTOR
0000001C 0000 298 F_DIAGNOSE=14*2 ;DIAGNOSE DRIVE
00000020 0000 299 F_WRITECHECK=20*2 ;WRITE CHECK DATA
0000002A 0000 300 F_WRITECHECKH=21*2 ;WRITE CHECK HEADER AND DATA
00000030 0000 301 F_WRITEDATA=24*2 ;WRITE DATA
00000032 0000 302 F_WRITEHEAD=25*2 ;WRITE HEADER AND DATA
00000034 0000 303 F_WRIETRACKD=26*2 ;WRITE TRACK DESCRIPTOR
00000038 0000 304 F_READDATA=28*2 ;READ DATA
0000003A 0000 305 F_READHEAD=29*2 ;READ HEADER AND DATA
0000003C 0000 306 F_READTRACKD=30*2 ;READ TRACK DESCRIPTOR
00000000 0000 307 F_AVAILABLE=F_NOP ;AVAILABLE
0000 308
0000 309
0000 310 : LOCAL DATA
0000 311
0000 312 : DRIVER PROLOGUE TABLE
0000 313 :
0000 314
0000 315 DPTAB - ;DEFINE DRIVER PROLOGUE TABLE
0000 316 END=DR_END,- ;END OF DRIVER
0000 317 ADAPTER=MBA,- ;ADAPTER TYPE
0000 318 FLAGS=DPTSM_SVP,- ;SYSTEM PAGE TABLE ENTRY REQUIRED
0000 319 UCBSIZE=UCBSK_DR_LENGTH,- ;UCB size
0000 320 NAME=DRDRIVER ;DRIVER NAME
0038 321 DPT_STORE INIT ;CONTROL BLOCK INIT VALUES
0038 322 DPT_STORE DDB,DB$$_ACPD,L,<^A\F11> ;DEFAULT ACP NAME
003F 323 DPT_STORE DDB,DB$$_ACPD+3,B,DB$$_PACK ;ACP CLASS
0043 324 DPT_STORE UCB,UCB$_FIPL,B,8 ;FORK IPL
0047 325 DPT_STORE UCB,UCB$_DEVCHAR,L,- ;DEVICE CHARACTERISTICS
0047 326 <DEVSM_FOD- ;FILES ORIENTED
0047 327 :DEVSM_DIR- ;DIRECTORY STRUCTURED
0047 328 :DEVSM_AVL- ;AVAILABLE
0047 329 :DEVSM_ELG- ;ERROR LOGGING ENABLED
0047 330 :DEVSM_SHR- ;SHAREABLE
0047 331 :DEVSM_IDV- ;INPUT DEVICE
0047 332 :DEVSM_ODV- ;OUTPUT DEVICE
0047 333 :DEVSM_RND> ;RANDOM ACCESS
004E 334 DPT_STORE UCB,UCB$_DEVCHAR2,L,- ;DEVICE CHARACTERISTICS
004E 335 <DEVSM_NNM> ;PREFIX NAME WITH "node$"
0055 336 DPT_STORE UCB,UCB$_DEVCLASS,B,DC$$_DISK ;DEVICE CLASS
0059 337 DPT_STORE UCB,UCB$_DEVBUFSIZ,W,512 ;DEFAULT BUFFER SIZE
005E 338 DPT_STORE UCB,UCB$_DIPL,B,21 ;DEVICE IPL
0062 339 DPT_STORE UCB,UCB$_ERTCNT,B,8 ;ERROR RETRY COUNT
0066 340 DPT_STORE UCB,UCB$_ERTMAX,B,8 ;MAX ERROR RETRY COUNT
006A 341 DPT_STORE REINIT ;CONTROL BLOCK RE-INIT VALUES
006A 342 DPT_STORE DDB,DB$$_DDT,D,DR$$_DDT ;DDT ADDRESS
```



```
006F 343          DPT_STORE END          ;
0000 344
0000 345 :
0000 346 : DRIVER DISPATCH TABLE
0000 347 :
0000 348
0000 349          DDTAB  DR,-              ;DRIVER DISPATCH TABLE
0000 350          DR_STARTIO,-          ;START I/O OPERATION
0000 351          DR_UNSLNT,-          ;UNSOLICITED INTERRUPT
0000 352          DR_FUNC_TABLE,-      ;FUNCTION DECISION TABLE
0000 353          0,-                  ;CANCEL I/O ENTRY POINT
0000 354          DR_REGDUMP,-          ;REGISTER DUMP ROUTINE
0000 355          <<RM_EC2+4+4+MBASL_BCR+4+8>>+<<5+5+1>*4>>,- ;DIAG BUFF SIZE
0000 356          <<RM_EC2+4+4+MBASL_BCR+4+8>>+<1*4>+<EMBSL_DV_REGS_AV>>,- ;ERLG BUFF SI
0000 357          DR_UNIT_INIT          ;UNIT INITIALIZATION
0038 358
0038 359 :
0038 360 : DATA CHECK FUNCTION TRANSLATION TABLE
0038 361 :
0038 362
0038 363 CHECKTAB:
0A' 0038 364          .BYTE  CDF_WRITECHECK      ;WRITE DATA
0A' 0039 365          .BYTE  CDF_WRITECHECK      ;READ DATA
12' 003A 366          .BYTE  CDF_WRITECHECKH     ;WRITE HEADER AND DATA
12' 003B 367          .BYTE  CDF_WRITECHECKH     ;READ HEADER AND DATA
003C 368
003C 369 :
003C 370 : DRIVE TYPE DESCRIPTOR TABLE
003C 371 :
003C 372 DR_DTDESC:
0014 003C 373          .WORD  ^X14              ; RM03
06 003E 374          .BYTE  DTS_RM03
20 003F 375          .BYTE  32
05 0040 376          .BYTE  5
0337 0041 377          .WORD  823
00020260 0043 378          .LONG  823*5*32
24A4D003 0047 379          .LONG  ^X24A4D003
0000000F 004B 380 DR_DTDESCLEN=-DR_DTDESC
004B 381
0016 004B 382          .WORD  ^X16              ; RM80
0D 004D 383          .BYTE  DTS_RM80
1F 004E 384          .BYTE  31
0E 004F 385          .BYTE  14
022F 0050 386          .WORD  559
0003B3AE 0052 387          .LONG  559*14*31
24A4D050 0056 388          .LONG  ^X24A4D050
0017 005A 389          .WORD  ^X17
0F 005C 390          .BYTE  DTS_RM05
20 005D 391          .BYTE  32
13 005E 392          .BYTE  19
0337 005F 393          .WORD  823
0007A2A0 0061 394          .LONG  823*19*32
24A4D005 0065 395          .LONG  ^X24A4D005
0022 0069 396          .WORD  ^X22
07 006B 397          .BYTE  DTS_RP07
32 006C 398          .BYTE  50
20 006D 399          .BYTE  32
```



```
0276 006E 400 .WORD 630 ; 630 CYLINDERS
000F6180 0070 401 .LONG 630*32*50 ; MAXIMUM BLOCKS
24A50007 0074 402 .LONG ^X24A50007 ; MEDIA IDENT "DR RP07"
0000 0078 403
00000089 007A 404 .WORD 0 ; END OF TABLE
00000098 0089 405 .BLKB DR_DTDESCLEN ; SPARE DRIVE TYPE SLOT
00000098 0089 406 .BLKB DR_DTDESCLEN ; SPARE DRIVE TYPE SLOT
0098 407
0098 408
0098 409 : HARDWARE I/O FUNCTION CODE TABLE
0098 410 :
0098 411
0098 412 FTAB:
0098 413 GENF F_NOP ; NO OPERATION
0099 414 GENF F_NOP ; (NO UNLOAD FUNCTION)
009A 415 GENF F_SEEK ; SEEK CYLINDER
009B 416 GENF F_RECAL ; RECALIBRATE
009C 417 GENF F_DRVCLR ; DRIVE CLEAR
009D 418 GENF F_NOP ; (NO RELEASE PORT)
009E 419 GENF F_OFFSET ; OFFSET HEADS
009F 420 GENF F_RETCENTER ; RETURN HEADS TO CENTERLINE
00A0 421 GENF F_PACKACK ; PACK ACKNOWLEDGE
00A1 422 GENF F_SEARCH ; SEARCH FOR SECTOR
00A2 423 GENF F_WRITECHECK ; WRITE CHECK
00A3 424 GENF F_WRITEDATA ; WRITE DATA
00A4 425 GENF F_READDATA ; READ DATA
00A5 426 GENF F_WRITEHEAD ; WRITE HEADER AND DATA
00A6 427 GENF F_READHEAD ; READ HEADER AND DATA
00A7 428 GENF F_WRIETRACKD ; WRITE TRACK DESCRIPTOR
00A8 429 GENF F_READTRACKD ; READ TRACK DESCRIPTOR
00A9 430 GENF F_AVAILABLE ; AVAILABLE
00AA 431 GENF F_WRITECHECKH ; WRITE CHECK HEADER AND DATA
00AB 432 GENF F_READPRESET ; READ IN PRESET
00AC 433 GENF F_DIAGNOSE ; DIAGNOSE THE DRIVE
00AD 434 GENF F_SEARCHA ; SEARCH AHEAD OF SECTOR
00AE 435
00AE 436 :
00AE 437 : OFFSET TABLE
00AE 438 :
00AE 439
00AE 440 OFFTAB:
00 00AE 441 .BYTE 0 ; RETURN TO CENTERLINE
01 00AF 442 .BYTE ^X01 ; + OFFSET (BIT 0 = OFFSET FLAG)
81 00B0 443 .BYTE ^X81 ; - OFFSET (BIT 0 = OFFSET FLAG)
00 00B1 444 .BYTE 0 ; RETURN TO CENTERLINE
00000004 00B2 445 OFFSIZ=.-OFFTAB ; SIZE OF OFFSET TABLE
```



```
.SBTTL FUNCTION DECISION TABLE
00B2 447
00B2 448 :+
00B2 449 : RM03 FUNCTION DECISION TABLE
00B2 450 :-
00B2 451
00B2 452 DR_FUNCABLE:
00B2 453 FUNCTAB
00B2 454 <NOP,-
00B2 455 UNLOAD,-
00B2 456 SEEK,-
00B2 457 RECAL,-
00B2 458 DRVCLR,-
00B2 459 RELEASE,-
00B2 460 OFFSET,-
00B2 461 RETCENTER,-
00B2 462 PACKACK,-
00B2 463 SEARCH,-
00B2 464 READPRESET,-
00B2 465 SENSECHAR,-
00B2 466 SETCHAR,-
00B2 467 SENSEMODE,-
00B2 468 SETMODE,-
00B2 469 WRITECHECK,-
00B2 470 WRITEHEAD,-
00B2 471 READHEAD,-
00B2 472 WRITETRACKD,-
00B2 473 READTRACKD,-
00B2 474 WRITECHECKH,-
00B2 475 DIAGNOSE,-
00B2 476 READLBLK,-
00B2 477 WRITELBLK,-
00B2 478 READPBLK,-
00B2 479 WRITEPBLK,-
00B2 480 READVBLK,-
00B2 481 WRITEVBLK,-
00B2 482 AVAILABLE,-
00B2 483 ACCESS,-
00B2 484 ACPCONTROL,-
00B2 485 CREATE,-
00B2 486 DEACCESS,-
00B2 487 DELETE,-
00B2 488 MODIFY,-
00B2 489 MOUNT>
00BA 490 FUNCTAB
00BA 491 <NOP,-
00BA 492 UNLOAD,-
00BA 493 SEEK,-
00BA 494 RECAL,-
00BA 495 DRVCLR,-
00BA 496 RELEASE,-
00BA 497 OFFSET,-
00BA 498 RETCENTER,-
00BA 499 PACKACK,-
00BA 500 SEARCH,-
00BA 501 READPRESET,-
00BA 502 DIAGNOSE,-
00BA 503 SENSECHAR,-

:FUNCTION DECISION TABLE
:LEGAL FUNCTIONS
:NO OPERATION
:UNLOAD VOLUME
:SEEK CYLINDER
:RECALIBRATE
:DRIVE CLEAR
:RELEASE PORT
:OFFSET HEADS
:RETURN HEADS TO CENTERLINE
:PACK ACKNOWLEDGE
:SEARCH FOR SECTOR
:READ IN PRESET
:SENSE CHARACTERISTICS
:SET CHARACTERISTICS
:SENSE MODE
:SET MODE
:WRITE CHECK
:WRITE HEADER AND DATA
:READ HEADER AND DATA
:WRITE TRACK DESCRIPTOR
:READ TRACK DESCRIPTOR
:WRITE CHECK HEADER AND DATA
:DIAGNOSE THE DRIVE
:READ LOGICAL BLOCK
:WRITE LOGICAL BLOCK
:READ PHYSICAL BLOCK
:WRITE PHYSICAL BLOCK
:READ VIRTUAL BLOCK
:WRITE VIRTUAL BLOCK
:AVAILABLE
:ACCESS FILE AND/OR FIND DIRECTORY ENTRY
:ACP CONTROL FUNCTION
:CREATE FILE AND/OR CREATE DIRECTORY ENTRY
:DEACCESS FILE
:DELETE FILE AND/OR DIRECTORY ENTRY
:MODIFY FILE ATTRIBUTES
:MOUNT VOLUME
:BUFFERED I/O FUNCTIONS
:NO OPERATION
:UNLOAD VOLUME
:SEEK CYLINDER
:RECALIBRATE
:DRIVE CLEAR
:RELEASE PORT
:OFFSET HEADS
:RETURN HEADS TO CENTERLINE
:PACK ACKNOWLEDGE
:SEARCH FOR SECTOR
:READ IN PRESET
:DIAGNOSE DRIVE
:SENSE CHARACTERISTICS
```



00BA	504	SETCHAR,-	:SET CHARACTERISTICS
00BA	505	SENSEMODE,-	:SENSE MODE
00BA	506	SETMODE,-	:SET MODE
00BA	507	AVAILABLE,-	:AVAILABLE
00BA	508	ACCESS,-	:ACCESS FILE AND/OR FIND DIRECTORY ENTRY
00BA	509	ACPCONTROL,-	:ACP CONTROL FUNCTION
00BA	510	CREATE,-	:CREATE FILE AND/OR CREATE DIRECTORY ENTRY
00BA	511	DEACCESS,-	:DEACCESS FILE
00BA	512	DELETE,-	:DELETE FILE AND/OR DIRECTORY ENTRY
00BA	513	MODIFY,-	:MODIFY FILE ATTRIBUTES
00BA	514	MOUNT>	:MOUNT VOLUME
00C2	515	FUNCTAB +ACPSREADBLK,-	:READ FUNCTIONS
00C2	516	<READTRACKD,-	:READ TRACK DESCRIPTOR
00C2	517	READHEAD,-	:READ HEADER
00C2	518	READLBLK,-	:READ LOGICAL BLOCK
00C2	519	READPBLK,-	:READ PHYSICAL BLOCK
00C2	520	READVBLK>	:READ VIRTUAL BLOCK
00CE	521	FUNCTAB +ACPSWRITEBLK,-	:WRITE FUNCTIONS
00CE	522	<WRITETRACKD,-	:WRITE TRACK DESCRIPTOR
00CE	523	WRITECHECK,-	:WRITE CHECK
00CE	524	WRITECHECKH,-	:WRITE CHECK HEADER AND DATA
00CE	525	WRITEHEAD,-	:WRITE HEADER
00CE	526	WRITELBLK,-	:WRITE LOGICAL BLOCK
00CE	527	WRITEPBLK,-	:WRITE PHYSICAL BLOCK
00CE	528	WRITEVBLK>	:WRITE VIRTUAL BLOCK
00DA	529	FUNCTAB +ACPSACCESS,<ACCESS,CREATE>	:ACCESS AND CREATE FILE OR DIRECTORY
00E6	530	FUNCTAB +ACPSDEACCESS,<DEACCESS>	:DEACCESS FILE
00F2	531	FUNCTAB +ACPSMODIFY,-	:
00F2	532	<ACPCONTROL,-	:ACP CONTROL FUNCTION
00F2	533	DELETE,-	:DELETE FILE OR DIRECTORY ENTRY
00F2	534	MODIFY>	:MODIFY FILE ATTRIBUTES
00FE	535	FUNCTAB +ACPSMOUNT,<MOUNT>	:MOUNT VOLUME
010A	536	FUNCTAB +EXESLCLDSKVALID,-	:LOCAL DISK VALID FUNCTIONS
010A	537	<UNLOAD,-	:UNLOAD VOLUME
010A	538	AVAILABLE,-	:UNIT AVAILABLE
010A	539	PACKACK>	:PACK ACKNOWLEDGE
0116	540	FUNCTAB +EXESZEROPARM,-	:ZERO PARAMETER FUNCTIONS
0116	541	<NOP,-	:NO OPERATION
0116	542	UNLOAD,-	:UNLOAD VOLUME
0116	543	RECAL,-	:RECALIBRATE
0116	544	DRVCLR,-	:DRIVE CLEAR
0116	545	RELEASE,-	:RELEASE PORT
0116	546	RETCENTER,-	:RETURN HEADS TO CENTERLINE
0116	547	READPRESET,-	:READ IN PRESET
0116	548	PACKACK,-	:PACK ACKNOWLEDGE
0116	549	AVAILABLE>	:AVAILABLE
0122	550	FUNCTAB +EXESONEPARM,-	:ONE PARAMETER FUNCTIONS
0122	551	<SEEK,-	:SEEK CYLINDER
0122	552	OFFSET,-	:OFFSET HEADS
0122	553	SEARCH,-	:SEARCH FOR SECTOR
0122	554	DIAGNOSE>	:DIAGNOSE THE DRIVE
012E	555	FUNCTAB +EXESSENSEMODE,-	:
012E	556	<SENSECHAR,-	:SENSE CHARACTERISTICS
012E	557	SENSEMODE>	:SENSE MODE
013A	558	FUNCTAB +EXESSETCHAR,-	:
013A	559	<SETCHAR,-	:SET CHARACTERISTICS
013A	560	SETMODE>	:SET MODE



```
0146 562 .SBTTL START I/O OPERATION
0146 563 :+
0146 564 : DR_STARTIO - START I/O OPERATION ON DEVICE UNIT
0146 565 :
0146 566 : THIS ENTRY POINT IS ENTERED TO START AN I/O OPERATION ON A DEVICE UNIT.
0146 567 :
0146 568 : INPUTS:
0146 569 :
0146 570 : R3 = ADDRESS OF I/O PACKET.
0146 571 : R5 = UCB ADDRESS OF DEVICE UNIT.
0146 572 :
0146 573 : OUTPUTS:
0146 574 :
0146 575 : FUNCTION DEPENDENT PARAMETERS ARE STORED IN THE DEVICE UCB, THE ERROR
0146 576 : RETRY COUNT IS RESET, AND THE FUNCTION IS EXECUTED. AT FUNCTION COMPLETION
0146 577 : THE OPERATION IS TERMINATED THROUGH REQUEST COMPLETE.
0146 578 :-
0146 579 :
0146 580 DR_STARTIO: ;START I/O OPERATION
0146 581 MOVB UCBSB_ERTMAX(R5),UCBSB_ERTCNT(R5) ;INITIALIZE ERROR RETRY COUNT
0080 C5 0081 C5 90 0146 581 MOVW IRPSW_FUNC(R3),UCBSW_FUNC(R5) ;SAVE FUNCTION CODE AND MODIFIERS
009A C5 20 A3 B0 0146 582 CLRW UCBSW_DR_MR(R5) ;CLEAR THE MAINTENANCE VALUE
00D4 C5 00D2 C5 B4 0153 583 BICW #^CDR_M_DUALPORT, - ;Clear software status and error log
FFF7 8F AA 0157 584 UCBSB_DR_SSTS(R5) ;bytes, except for dualport bit.
50 38 A3 D0 015E 585 MOVL IRPSL_MEDIA(R3),R0 ;GET PARAMETER LONGWORD
0162 586 :
0162 587 :
0162 588 : MOVE FUNCTION DEPENDENT PARAMETERS TO UCB
0162 589 :
0162 590 :
0162 591 :
0162 592 10$: EXTZV #IRPSV_FCODE,#IRPSS_FCODE,- ;EXTRACT I/O FUNCTION CODE
51 06 00 EF 0162 592 IRPSW_FUNC(R3),R1 ;
51 20 A3 91 0165 593 CMPB #IOS_SEEK,R1 ;SEEK FUNCTION?
51 02 91 0168 594 BEQL 20$ ;IF EQL YES
51 1E 13 016B 595 CMPB #IOS_OFFSET,R1 ;OFFSET FUNCTION?
51 06 91 016D 596 BEQL 30$ ;IF EQL YES
51 20 13 0170 597 CMPB #IOS_SEARCH,R1 ;SEARCH FUNCTION?
51 09 91 0172 598 BEQL 40$ ;IF EQL YES
51 22 13 0175 599 CMPB #IOS_DIAGNOSE,R1 ;DIAGNOSE FUNCTION?
51 1D 91 0177 600 BEQL 45$ ;IF EQL YES
U0BC C5 50 D0 017A 601 MOVL R0,UCBSW_DA(R5) ;STORE PARAMETER LONGWORD
51 18 91 0181 602 CMPB #IOS_WRITECHECKH,R1 ;DISJOINT FUNCTION CODE?
51 22 1A 0184 604 BGTRU 50$ ;IF GTRU NO
51 06 A2 0186 605 SUBW #IOS_WRITECHECKH-IOS_AVAILABLE-1,R1 ;MAKE FUNCTION TABLE INDEX
1D 11 0189 606 BRB 50$ ;
018B 607 :
018B 608 :
018B 609 : SEEK FUNCTION - SET CYLINDER ADDRESS
018B 610 :
018B 611 :
00BE C5 50 B0 018B 612 20$: MOVW R0,UCBSW_DC(R5) ;SET CYLINDER ADDRESS
51 16 11 0190 613 BRB 50$ ;
0192 614 :
0192 615 :
0192 616 : OFFSET FUNCTION - SET CURRENT OFFSET VALUE
0192 617 :
0192 618 :
```



```
00C8 C5 50 90 0192 619 30$: MOVB R0,UCBSW_OFFSET(R5) ;SET OFFSET VALUE
OF 11 0197 620 BRB 50$
0199 621
0199 622
0199 623 : SEARCH FUNCTION - SET SECTOR ADDRESS
0199 624
0199 625
00BC C5 50 90 0199 626 40$: MOVB R0,UCBSW_DA(R5) ;SET SECTOR ADDRESS
08 11 019E 627 BRB 50$
01A0 628
01A0 629
01A0 630 : DIAGNOSE FUNCTION - SET MAINTENANCE VALUE
01A0 631
01A0 632
00D2 C5 50 B0 01A0 633 45$: MOVW R0,UCBSW_DR_MR(R5) ;SET MAINTENANCE VALUE
51 03 A2 01A5 634 SUBW #IOS_DIAGNOSE-IOS_READPRESET-1,R1 ;MAKE A FUNCTION TABLE INDEX
01A8 635
01A8 636
01A8 637 : FINISH PREPROCESSING
01A8 638
01A8 639
0092 C5 51 90 01A8 640 50$: MOVB R1,UCBSB_FEX(R5) ;SAVE FUNCTION DISPATCH INDEX
54 24 A5 D0 01AD 641 MOVL UCBSL_CRB(R5),R4 ;GET ADDRESS OF CRB
54 2C B4 D0 01B1 642 MOVL @CRBSL_INTD+VECSL_IDB(R4),R4 ;GET FIRST CONTROLLER CSR ADDRESS
00 68 A5 00 E4 01B5 643 BBS #UCBSV_ECC,UCBSW_DEVSTS(R5),FDISPATCH ;CLEAR ECC CORRECTION MADE
01BA 644
01BA 645
01BA 646 : CENTRAL FUNCTION DISPATCH
01BA 647
01BA 648
01BA 649 FDISPATCH: ;FUNCTION DISPATCH
53 58 A5 D0 01BA 650 MOVL UCBSL_IRP(R5),R3 ;RETRIEVE ADDRESS OF I/O PACKET
0D 2A A3 08 E0 01BE 651 BBS #IRPSV_PHYSIO,IRPSW_STS(R3),10$ ;IF SET, PHYSICAL I/O FUNCTION
08 64 A5 08 E0 01C3 652 BBS #UCBSV_VALID,UCBSW_STS(R5),10$ ;IF SET, VOLUME SOFTWARE VALID
50 0254 8F 3C 01C8 653 MOVZWL #SS$ VOLINV,R0 ;SET VOLUME INVALID STATUS
06D1 31 01CD 654 BRW RESETXFR
01D0 655
01D0 656 : UNIT IS SOFTWARE VALID OR FUNCTION IS PHYSICAL I/O
01D0 657
01D0 658
01D0 659
50 0092 C5 9A 01D0 660 10$: MOVZBL UCBSB_FEX(R5),R0 ;GET DISPATCH FUNCTION CODE
00C9 C5 10 90 01D5 661 MOVB #RM_OF_M_FMT/256,UCBSW_OFFSET+1(R5) ;CLEAR ECI, HCI, AND SET FORMAT
00CB C5 01 90 01DA 662 MOVB #1,UCBSB_OFFRTC(R5) ;SET INITIAL OFFSET RETRY COUNT
00CA C5 94 01DF 663 CLRB UCBSB_OFFNDX(R5) ;CLEAR INITIAL OFFSET TABLE INDEX
01E3 664
01E3 665 : CHECK FOR DIAGNOSTIC MODIFIERS
01E3 666
01E3 667
2F 2A A3 08 E1 01E3 668 BBC #IRPSV_PHYSIO,IRPSW_STS(R3),40$ ;IF CLEAR, NOT PHYSICAL I/O
06 009A C5 06 E1 01E8 669 BBC #IOSV_COMMOD,UCBSW_FUNC(R5),15$ ;IF CLEAR, NO COMMAND MODIFIER
00C9 C5 80 8F 88 01EE 670 BISB #RM_OF_M_CMO/256,UCBSW_OFFSET+1(R5) ;SET COMMAND MODIFIER
01F4 671
06 009A C5 07 E1 01F4 672 15$: BBC #IOSV_MOVETRACKD,UCBSW_FUNC(R5),20$ ;IF CLR, NO MOVE TRACK DESC
00C9 C5 40 8F 88 01FA 673 BISB #RM_OF_M_MTD/256,UCBSW_OFFSET+1(R5) ;SET MOVE TRACK DESCRIPTOR
0200 674
06 009A C5 08 E1 0200 675 20$: BBC #IOSV_DIAGNOSTIC,UCBSW_FUNC(R5),30$ ;IF CLEAR, NOT DIAG MODE
```



```
00D3 C5 80 8F 88 0206 676 BISB #RM_MR_M_DM/256,UCBSW_DR_MR+1(R5) ;SET DIAGNOSTIC MODE
05 009A C5 09 E1 020C 677
00C9 C5 02 88 020C 678 30$: BBC #IOSV SKPSECINH,UCBSW_FUNC(R5),40$ ;IF CLEAR, NO SSEI MODIFIER
0212 679 BISB #RM_OF_M_SSEI/256,UCBSW_OFFSET+1(R5) ;SET SKIP SECTOR ERR INH
0217 680
0217 681 ; DISPATCH TO FUNCTION HANDLING ROUTINE
0217 682
0217 683
0217 684 40$:
0217 685 CASE RO,<- ;DISPATCH TO FUNCTION HANDLING ROUTINE
0217 686 NOP,- ;NO OPERATION
0217 687 UNLOAD,- ;UNLOAD VOLUME
0217 688 SEEK,- ;SEEK CYLINDER
0217 689 RECAL,- ;RECALIBRATE
0217 690 DRVCLR,- ;DRIVE CLEAR
0217 691 RELEASE,- ;RELEASE PORT
0217 692 OFFSET,- ;OFFSET HEADS
0217 693 RETCENTER,- ;RETURN HEADS TO CENTER
0217 694 PACKACK,- ;PACK ACKNOWLEDGE
0217 695 SEARCH,- ;SEARCH FOR SECTOR
0217 696 WRITECHECK,- ;WRITE CHECK DATA
0217 697 WRITEDATA,- ;WRITE DATA
0217 698 READDATA,- ;READ DATA
0217 699 WRITEHEAD,- ;WRITE HEADER AND DATA
0217 700 READHEAD,- ;READ HEADER AND DATA
0217 701 WRITETRACKD,- ;WRITE TRACK DESCRIPTOR
0217 702 READTRACKD,- ;READ TRACK DESCRIPTOR
0217 703 AVAILABLE,- ;AVAILABLE
0217 704 WRITECHECKH,- ;WRITE CHECK HEADER AND DATA
0217 705 READPRESET,- ;READIN PRESET
0217 706 DIAGNOSE> ;DIAGNOSE DRIVE
0245 707
0245 708 ; UNLOAD or AVAILABLE - Clear UCBSV VALID
0245 709 ; This is the only operation which these functions need to perform. All
0245 710 ; devices supported by this driver do not have an unload function, and the
0245 711 ; available function code should only clear the UCBSV_VALID bit.
0245 712
0245 713 UNLOAD:
0245 714 AVAILABLE:
64 A5 0800 8F AA 0245 715 BICW #UCBSM_VALID, UCBSW_STS(R5) ;Clear the software volume valid
00AF 31 024B 716 BRW NORMAL ;bit and complete function.
024E 717
024E 718 ;
024E 719 ; PACKACK - Set UCBSV_VALID and proceed with hardware pack acknowledge
024E 720 ; function
024E 721 ;
024E 722 PACKACK:
64 A5 0800 8F AB 024E 723 BISW #UCBSM_VALID, UCBSW_STS(R5) ;Set the software volume valid
0254 724 ; BRB NOP ;bit and complete function.
0254 725
0254 726 ;
0254 727 ; NO OPERATION, SEEK, RECALIBRATE, DRIVE CLEAR, RELEASE, OFFSET,
0254 728 ; RETURN TO CENTER LINE, SEARCH, AND READ IN PRESET
0254 729 ;
0254 730
0254 731 NOP: ;NO OPERATION
0254 732 SEEK: ;SEEK CYLINDER
```



```
0254 733 RECAL: ;RECALIBRATE
0254 734 DRVCLR: ;DRIVE CLEAR
0254 735 RELEASE: ;RELEASE PORT
0254 736 OFFSET: ;OFFSET READ HEADS
0254 737 RETCENTER: ;RETURN TO CENTERLINE
0254 738 SEARCH: ;SEARCH FOR SECTOR
0254 739 READPRESET: ;READIN PRESET
0254 740 EXFUNC RETRY ;EXECUTE HOUSEKEEPING FUNCTION
00A1 31 0259 741 BRW NORMAL ;
025C 742 ;
025C 743 ;
025C 744 ; WRITE TRACK DESCRIPTOR and READ TRACK DESCRIPTOR
025C 745 ; Both want to SEEK rather than to SEARCH to arrive on cylinder.
025C 746 ;
025C 747 ;
025C 748 WRITETRACKD: ;WRITE TRACK DESCRIPTOR
00D4 C5 04 88 025C 749 BISB #DR_M_NOECC, UCBSB_DR_SSIS(R5) ; Signal don't correct with ECC.
0261 750 ;
0261 751 READTRACKD: ;READ TRACK DESCRIPTOR
23 009A 0C E0 0261 752 BBS #IOSV_INHSEEK, -
0263 753 UCBSW_FUNC(R5), TRANRQCH ; If set, NO explicit SEEK
0267 754 EXFUNC RETRY, F SEEK ; Seek to cylinder
19 11 026F 755 BRB TRANRQCH ; and branch around to common code.
0271 756 ;
0271 757 ;
0271 758 ; WRITE CHECK DATA AND WRITE CHECK HEADER AND DATA
0271 759 ;
0271 760 ;
0271 761 WRITECHECK: ;WRITE CHECK DATA
0271 762 WRITECHECKH: ;WRITE CHECK HEADER AND DATA
00 009A C5 0E E4 0271 763 BBSC #IOSV_DATACHECK, UCBSW_FUNC(R5), WRITEDATA ; CLEAR DATA CHECK REQUEST
0277 764 ;
0277 765 ;
0277 766 ; WRITE DATA, WRITE HEADER AND DATA,
0277 767 ; WRITE CHECK DATA, AND WRITE CHECK HEADER AND DATA
0277 768 ;
0277 769 ;
0277 770 WRITEDATA: ;WRITE DATA
0277 771 WRITEHEAD: ;WRITE HEADER AND DATA
00D4 C5 04 88 0277 772 BISB #DR_M_NOECC, UCBSB_DR_SSIS(R5) ; Signal don't correct with ECC.
027C 773 ;
027C 774 ;
027C 775 ; READ DATA, READ HEADER AND DATA,
027C 776 ; WRITE DATA, WRITE HEADER AND DATA,
027C 777 ; WRITE CHECK DATA, AND WRITE CHECK HEADER AND DATA
027C 778 ;
027C 779 ;
027C 780 READDATA: ;READ DATA
027C 781 READHEAD: ;READ HEADER AND DATA
08 009A C5 0C E0 027C 782 BBS #IOSV_INHSEEK, UCBSW_FUNC(R5), TRANRQCH ; IF SET, NO EXPLICIT SEEK
0282 783 EXFUNC RETRY, F_SEARCHA ; SEARCH AHEAD OF STARTING SECTOR
028A 784 ;
028A 785 ;
028A 786 ; DATA TRANSFER OR DIAGNOSE - REQUEST CHANNEL
028A 787 ;
028A 788 ;
028A 789 DIAGNOSE: ;DIAGNOSE
```



```
028A 790 TRANRQCH: ;DATA TRANSFER
028A 791 REQPCAN LOW ;REQUEST PRIMARY CHANNEL
0290 792
0290 793 :
0290 794 : DATA TRANSFER - CHANNEL ALREADY OWNED
0290 795 :
0290 796
0290 797 TRANNOCH: ;DATA TRANSFER CHANNEL OWNED
50 0092 C5 9A 0290 798 MOVZBL UCBSB_FEX(R5),R0 ;GET FUNCTION DISPATCH INDEX
0295 799 EXFUNC TRANXT ;EXECUTE TRANSFER FUNCTION
029A 800
029A 801 :
029A 802 : DATA CHECK
029A 803 :
029A 804
029A 805 DATACHECK: ;DATA CHECK
5D 009A C5 0E E1 029A 806 BBC #IOSV_DATACHECK,UCBSW_FUNC(R5),NORMAL ;IF CLR, NO DATA CHECK
50 0639 8F 3C 02A0 807 MOVZWL #SS$_BASECC,R0 ;ASSUME ECC CORRECTION WAS MADE
56 68 A5 00 E0 02A5 808 BBS #UCBSV_ECC,UCBSW_DEVSTS(R5),CHECKXT ;IF SET, ECC CORRECTION MADE
00D4 C5 01 88 02AA 809 RELCHAN ;RELEASE CHANNEL
00C9 C5 10 90 02B0 810 BISB #DR_M_DCK,UCBSB_DR_SSTS(R5) ;SET DATA CHECK IN PROGRESS
00D4 C5 04 88 02B5 811 MOVB #RM_OF_M_FMT/256,UCBSW_OFFSET+1(R5) ;CLEAR ECI, HCI, AND SET FORMAT
00CB C5 01 90 02BA 812 BISB #DR_M_NOECC,UCBSB_DR_SSTS(R5) ;Signal don't correct with ECC.
00CA C5 94 02BF 813 MOVB #1,UCBSB_OFFRTC(R5) ;SET INITIAL OFFSET RETRY COUNT
52 58 A5 D0 02C4 814 CLRB UCBSB_OFFNDX(R5) ;CLEAR INITIAL OFFSET TABLE INDEX
78 A5 2C A2 7D 02C8 815 MOVL UCBSL_IRP(R5),R2 ;GET ADDRESS OF IRP
00BC C5 38 A2 D0 02CC 816 MOVQ IRPSL_SVAPTE(R2),UCBSL_SVAPTE(R5) ;RESET TRANSFER PARAMETERS
0B 2A A2 08 E1 02D7 817 MOVL IRPSL_MEDIA(R2),UCBSW_BA(R5) ;
05 009A C5 09 E1 02DC 818 BBC #IRPSV_PHYSIO,IRPSW_STS(R2),CHECKRETRY ;IF CLEAR NOT PHYS I/O
00C9 C5 02 88 02E2 819 BBC #IOSV_SKIPSECTINH,UCBSW_FUNC(R5),CHECKRETRY ;IF CLEAR NO SSEI MOD
02E7 820 BISB #RM_OF_M_SSEI/256,UCBSW_OFFSET+1(R5) ;SET SKIP SECTOR ERR INH
02E7 821
02E7 822 :
02E7 823 : DATA CHECK RETRY
02E7 824 :
02E7 825
02E7 826 CHECKRETRY: ;DATA CHECK RETRY
50 0092 C5 9A 02E7 827 REQPCAN LOW ;REQUEST PRIMARY CHANNEL FOR DATA CHECK
50 FD36 CF40 9A 02ED 828 MOVZBL UCBSB_FEX(R5),R0 ;GET FUNCTION DISPATCH INDEX
02F2 829 MOVZBL CHECKTAB-CDF_WRITEDATA[R0],R0 ;GET CASE TABLE INDEX
02F8 830 EXFUNC TRANXT ;EXECUTE DATA CHECK FUNCTION
02FD 831
02FD 832 :
02FD 833 : SUCCESSFUL OPERATION COMPLETION
02FD 834 :
02FD 835
02FD 836 NORMAL:
50 01 3C 02FD 837 MOVZWL #SS$_NORMAL,R0 ;SET NORMAL COMPLETION STATUS
0300 838 CHECKXT:
0208 31 0300 839 BRW FUNCXT ;
0303 840
0303 841 :
0303 842 : TRANSFER ENDED WITH A RETRIABLE ERROR
0303 843 :
0303 844
0303 845 TRANXT: ;TRANSFER EXIT
0093 C5 0B 91 0303 846 CMPB #CDF_WRITEDATA,UCBSB_CEX(R5) ;WRITE DATA FUNCTION?
```



```
0093 C5 24 13 0308 847 BEQL RETRY ;IF EQL YES
OD 91 030A 848 CMPB #CDF_WRITEHEAD,UCBSB_CEX(R5);WRITE HEADER FUNCTION?
1D 13 030F 849 BEQL RETRY ;IF EQL YES
51 00064F74 8F D3 0311 850 BITL #MBASH_SR_DLT!- ;DATA LATE OR,
0318 851 MBASH_SR_INVMAP!- ;INVALID MAP REGISTER OR,
0318 852 MBASH_SR_MAPPE!- ;MAP REGISTER PARITY ERROR OR,
0318 853 MBASH_SR_MCPE!- ;MASSBUS CONTROL PARITY ERROR OR,
0318 854 MBASH_SR_SPE!- ;MBA SILO PARITY ERROR OR,
0318 855 MBASH_SR_MDPE!- ;MASSBUS DATA PARITY ERROR OR,
0318 856 MBASH_SR_MXF!- ;MISSED TRANSFER OR,
0318 857 MBASH_SR_NED!- ;NONEXISTENT DISK OR,
0318 858 MBASH_SR_RDS!- ;READ DATA SUBSTITUTE OR,
0318 859 MBASH_SR_WCKLWR!- ;WRITE CHECK LOWER BYTE OR,
0318 860 MBASH_SR_WCKUPR,R1 ;WRITE CHECK UPPER BYTE?
00D0 C5 14 12 0318 861 BNEQ RETRY ;IF NEQ YES - RETRY FUNCTION
1C88 8F B3 031A 862 BITW #RM_ER2_M_DPE!- ;DATA PARITY ERROR OR,
0321 863 RM_ER2_M_DVC!- ;DEVICE CHECK OR,
0321 864 RM_ER2_M_LBC!- ;LOSS OF BIT CLOCK OR,
0321 865 RM_ER2_M_LSC!- ;LOSS OF SYSTEM CLOCK OR,
0321 866 RM_ER2_M_IVC,UCBSW_DR_ER2(R5);INVALID COMMAND?
OA 52 0B 12 0321 867 BNEQ RETRY ;IF NEQ YES - RETRY FUNCTION
08 08 E0 0323 868 BBS #RM_ER1_V_HCRC,R2,ECC ;Test HCRC before HCE.
52 20A8 8F B3 0327 869 BITW #RM_ER1_M_OPI!- ;OPERATION INCOMPLETE OR,
032C 870 RM_ER1_M_PAR!- ;PARITY ERROR OR,
032C 871 RM_ER1_M_HCE!- ;HEADER COMPARE ERROR OR,
032C 872 RM_ER1_M_WCF,R2 ;WRITE CLOCK FAIL?
03 13 032C 873 BEQL ECC ;IF EQL NO
0110 31 032E 874 RETRY: BRW RETRYERR ;RETRIABLE ERROR
032E 875
0331 876
0331 877 ; ECC, DRIVE TIMING, OR HEADER ERROR - APPLY ECC OR PERFORM OFFSET RECOVERY
0331 878
0331 879
0331 880
0331 881 ECC:
51 7E A5 00D8 C5 A1 0331 882 ADDW3 UCBSL_DR_BCR(R5), - ;ECC CORRECTION
0338 883 UCBSW_BCNT(R5), R1 ;Compute bytes transfered then
50 51 FFFF01FF 8F CB 0338 884 BICL3 #*XFFF01FF, R1, R0 ;clear byte offset bits and
77 13 0340 885 BEQL OFF ;convert result to a longword.
51 01FF 8F B3 0342 886 BITW #*X1FF, R1 ;Branch if whole blocks xfered is zero.
70 12 0347 887 BNEQ OFF ;Was a partial block transfered?
10 52 08 E1 0349 888 BBC #RM_ER1_V_HCRC, R2, 10$ ;Branch if partial block transfered.
07 91 034D 889 CMPB #DTS_RP07,- ;Branch if error was not HCRC.
41 A5 034F 890 UCBSB_DEVTYPE(R5) ;Is this drive an RP07?
11 12 0351 891 BNEQ 20$ ;Branch if not.
00000400 8F E1 0353 892 BBC #RM_OF_M_HCI,- ;Branch if header compare inhibit
07 00C8 C5 0359 893 ;isn't set.
50 00000200 8F C2 035D 894 10$: SUBL2 #512,-R0 ;Else, reduce bytes xfered by a block.
52 1140 8F B3 0364 895 20$: BITW #RM_ER1_M_DTE!- ;For: DRIVE TIMING ERROR
0369 896 RM_ER1_M_ECH!- ;ECC HARD ERROR
0369 897 RM_ER1_M_HCRC,R2 ;HEADER CRC ERROR
48 00D4 C5 4E 12 0369 898 BNEQ OFF ;perform offset recovery.
02 E0 036B 899 BBS #DR_V_NOECC, - ;If it won't help, skip ECC correction.
0371 900 UCBSB_DR_SSIS(R5), OFF
52 00C6 C5 7E 52 7D 0371 901 MOVQ R2,-(SP) ;Save work registers.
00 EA 0374 902 FFS #0,#11,UCBSW_EC2(R5),R2 ;Find the first error bit in the ECC
037B 903 ;pattern.
```



```
53 0A 52 C3 037B 904          SUBL3  R2,#10,R3          ; Get the number of error bits
                                037F 905                ; remaining in the pattern.
                                09 15 037F 906          BLEQ  30$          ; Branch if no other bits in pattern.
52 00C6 C5 53 52 D6 0381 907          INCL  R2          ; Point to next bit in pattern.
                                52 EF 0383 908          EXTZV R2,R3,UCBSW_EC2(R5),R2 ; Is there more than one error bit set?
                                0C BA 038A 909 30$:      POPR  #M<R3,R2>      ; Restore work registers without
                                038C 910                ; affecting flags.
                                26 1A 038C 911          BGTRU  DEFER_ECC      ; If more than one error bit set, don't
                                038E 912                ; apply ECC correction.
                                038E 913                ;
                                038E 914                ; APPLY_ECC -
                                038E 915                ;
                                038E 916                ; Apply ECC correction to correct a single bit error.
                                038E 917                ;
                                038E 918                ;
                                038E 919 APPLY_ECC:
                                7E 51 3C 038E 920      MOVZWL R1, -(SP)      ; Save total bytes transfered, inc. ECC.
                                00000000'GF 16 0391 921      JSB   G^IOC$APPLYECC    ; Apply ECC correction.
                                50 8ED0 0397 922      POPL  R0          ; Retrieve transfered byte count.
                                00000000'GF 16 039A 923      JSB   G^IOC$UPDATRANSF  ; Update transfer parameters.
                                00CA C5 94 03A0 924      CLRB  UCBSB_OFFNDX(R5)    ; Reset offset table index.
                                02 8A 03A4 925      BICB  #DR_M_OM,-          ; Clear offset mode.
                                00D4 C5 03A6 926      UCBSB_DR_SSTS(R5)
                                7E A5 B5 03A9 927      TSTW  UCBSW_BCNT(R5)
                                03 13 03AC 928      BEQL  20$          ; Any more to transfer?
                                FEDF 31 03AE 929      BRW   TRNNOCH          ; If EQL no.
                                FEE6 31 03B1 930 20$:     BRW   DATACHECK        ; Transfer next segment.
                                03B4 931                ; Check for write check.
                                03B4 932                ;
                                03B4 933                ; DEFER_ECC -
                                03B4 934                ;
                                03B4 935                ; Don't apply ECC correction for multiple bit errors unless the error cannot
                                03B4 936                ; be recovered with offset retries.
                                03B4 937                ;
                                03B4 938                ;
                                03B4 939 DEFER_ECC:
                                00D4 10 88 03B4 940      BISB  #DR_M_ECC_DEFER,-    ; Set flag to indicate that ECC
                                03B6 941      UCBSB_DR_SSTS(R5)    ; can be used if offset recovery fails.
                                03B9 942                ;
                                03B9 943                ;
                                03B9 944                ; OFF - OFFSET RECOVERY
                                03B9 945                ;
                                03B9 946                ; THIS CODE IS EXECUTED WHEN A DRIVE TIMING ERROR, HEADER CRC, OR ECC
                                03B9 947                ; HARD ERROR IS DETECTED ON A READ FUNCTION.
                                03B9 948                ;
                                03B9 949                ;
                                50 D5 03B9 950 OFF:      ; OFFSET RECOVERY
                                33 13 03BB 951      TSTL  R0          ; ANY GOOD DATA TRANSFERED?
                                03BD 952      BEQL  30$          ; IF EQL NO
                                03BD 953                ;
                                03BD 954                ;
                                03BD 955                ; THE TRANSFER ENDED IN AN ERROR BUT THERE WERE SECTORS TRANSFERED THAT
                                03BD 956                ; CONTAINED GOOD DATA. SINCE THE ERROR COULD HAVE BEEN CAUSED BY A CYLIN-
                                03BD 957                ; DER CROSSING, THE GOOD DATA IS SAVED AND THE TRANSFER IS RETRIED FROM THE
                                03BD 958                ; POINT OF ERROR.
                                03BD 959                ;
                                03BD 960
```



```
00000000'GF 16 03BD 961 JSB G*IOCSUPDATRANSF ;UPDATE TRANSFER PARAMETERS
      7E A5 B5 03C3 962 TSTW UCB$W_BCNT(R5) ; Any more data to transfer?
      03 12 03C6 963 BNEQ 5$ ; Branch if so.
      FECF 31 03C8 964 BRW DATACHECK ; Otherwise, go check for write check.
00CA C5 94 03CB 965 5$: CLRB UCB$B_OFFNDX(R5) ;RESET OFFSET TABLE INDEX
00CB C5 10 90 03CF 966 10$: MOV B #16,UCB$B_OFFRTC(R5) ;SET OFFSET RETRY COUNT
00CA C5 04 91 03D4 967 CMPB #OFFSIZ,UCB$B_OFFNDX(R5) ;ALL OFFSETS TRIED?
      08 12 03D9 968 BNEQ 20$ ; Branch if not.
      04 E4 03DB 969 BBSC #DR_V_ECC_DEFER,- ; Correct the error with ECC if we can.
      00D4 C5 03DD 970 UCB$B_DR_SSTS(R5),-
      AD 03E0 971 APPLY_ECC
      53 11 03E1 972 BRB 90$ ; Otherwise, fatal error.
      00D4 C5 02 8A 03E9 974 20$: RELCHAN ;RELEASE CHANNEL
      35 11 03EE 975 BICB #DR_M_OM,UCB$B_DR_SSTS(R5) ;CLEAR OFFSET MODE
      03F0 976 BRB 60$
      03F0 977 ;
      03F0 978 ; NO GOOD DATA TRANSFERED - CHECK IF CHANGE IN OFFSET NEEDED
      03F0 979 ;
      03F0 980
52 9040 8F B3 03F0 981 30$: BITW #RM_ER1_M_DCK!- ;DATA CHECK OR,
      03F5 982 RM_ER1_M_DTE!- ;DRIVE TIMING OR,
      03F5 983 RM_ER1_M_ECH,R2 ;ECC HARD ERROR?
      03F5 984 BNEQ 40$ ;IF NEQ YES
00C9 C5 05 12 03F5 985 BISB #RM_OF_M_HCI/256,UCB$W_OFFSET+1(R5) ;SET HEADER COMPARE INHIBIT
      00CB C5 04 88 03F7 986 40$: DECB UCB$B_OFFRTC(R5) ;CHANGE CURRENT OFFSET?
      28 12 0400 987 BNEQ 70$ ;IF NEQ NO
      00CA C5 96 0402 988 INCB UCB$B_OFFNDX(R5) ;UPDATE OFFSET TABLE INDEX
50 00CA C5 9A 0406 989 MOVZBL UCB$B_OFFNDX(R5),R0 ;GET NEXT OFFSET TABLE INDEX
00CB C5 FC9D CF40 90 040B 990 MOV B OFFTAB-1[R0],UCB$W_OFFSET(R5) ;GET NEXT OFFSET VALUE
      BA 13 0413 991 BEQL 10$ ;IF EQL RETURN TO CENTERLINE
      00CB C5 02 90 0415 992 MOV B #2,UCB$B_OFFRTC(R5) ;SET OFFSET RETRY COUNT
      00D4 C5 02 88 0420 993 RELCHAN ;RELEASE CHANNEL
      00C9 C5 04 8A 0425 994 BISB #DR_M_OM,UCB$B_DR_SSTS(R5) ;SET OFFSET MODE
03 00D4 C5 00 E0 042A 995 60$: BICB #RM_OF_M_HCI/256,UCB$W_OFFSET+1(R5) ;CLEAR HEADER COMPARE INHIBIT
      FE57 31 0430 996 70$: BBS #DR_V_DCK,UCB$B_DR_SSTS(R5),80$ ;IF SET, DATA CHECK FUNCTION
      FEB1 31 0433 997 BRW TRANRQCH ;TRY FUNCTION AGAIN
      0436 998 80$: BRW CHECKRETRY ;TRY DATA CHECK AGAIN
      0436 999
      50 04 A3 D0 0436 1000 90$: MOVL RM_DS(R3),R0 ;GET DRIVE STATUS
51 00CC C5 D0 043A 1001 MOVL UCB$B_DR_SR(R5),R1 ;GET MBA STATUS
      32 11 043F 1002 BRB FATALERR
      0441 1003
      0441 1004 ;
      0441 1005 ; RETRIABLE ERROR
      0441 1006 ;
      0441 1007
      0441 1008 RETRYERR: ;RETRIABLE ERROR
      07 BB 0441 1009 PUSH R #*M<R0,R1,R2> ; Save error status registers.
      0443 1010 RELCHAN ; Release channel before possible RECAL
      07 BA 0449 1011 POP R #*M<R0,R1,R2> ; Restore error status registers.
04 00D0 C5 0E E0 044B 1012 BBS #RM_ER2_V_SKI,UCB$W_DR_ER2(R5),10$ ;IF SET, SEEK INCOMPLETE
      OD 52 07 E1 0451 1013 BBC #RM_ER1_V_HCE,R2,20$ ;IF CLR, HEADER COMPARED
      0455 1014 10$: EXFUNC FATALERR,F RECAL ;RECALIBRATE HEADS
      52 2000 8F 3C 045D 1015 MOVZWL #RM_ER1_M_OPI,R2 ;SET AN ERROR FOR CALLER TO SEE
      0080 C5 97 0462 1016 20$: DECB UCB$B_ERTCNT(R5) ;ANY RETRIES LEFT?
      0B 13 0466 1017 BEQL FATALERR ;IF EQL NO
```



```
FD47 31 0468 1018 EXFUNC FATALERR,F_DRVCLR ; Issue drive clear before retrying.
      0470 1019 BRW FDISPATCH ;
      0473 1020
      0473 1021 :
      0473 1022 : FATAL CONTROLLER/DRIVE ERROR, ERROR RETRY COUNT EXHAUSTED, ERROR RETRY
      0473 1023 : INHIBITED, OR FINAL OFFSET TRIED
      0473 1024 :
      0473 1025
      0473 1026 FATALERR:
      0473 1027 BBS #RM DS V MOL,R0,10$ ;FATAL ERROR - SET STATUS
      0477 1028 MOVZWL #SS$ MEDOFFL,R0 ; Branch if not offline.
      047C 1029 BRW FUNCXT ; Otherwise, set medium offline status
      047F 1030 10$: BBS #RM DS V V$,R0,20$ ; and branch to common completion exit.
      0483 1031 MOVZWL #SS$ VOLINV,R0 ; Branch if not volume invalid.
      0488 1032 BRW FUNCXT ; Otherwise, set volume invalid status.
      048B 1033 20$: BBS #RM ER1 V UNS,R2,30$ ; and branch to common completion exit.
      048F 1034 MOVZWL #SS$ UNSAFE,R0 ; Branch if not drive unsafe.
      0494 1035 BRW FUNCXT ; Otherwise, set drive unsafe status.
      0497 1036 30$: MOVZWL #SS$ OPINCOMPL,R0 ; and branch to common completion exit.
      049C 1037 BBS #RM ER1 V OPI,R2,FUNCXT ; SET OPERATION INCOMPLETE STATUS
      04A0 1038 MOVZWL #SS$ WRITCK,R0 ; IF SET, OPERATION INCOMPLETE
      04A5 1039 BBS #RM ER1 V WLE,R2,FUNCXT ; SET WRITE LOCK ERROR STATUS
      04A9 1040 MOVZWL #SS$ IVADDR,R0 ; IF SET, WRITE LOCK ERROR
      04AE 1041 BITW #RM ER1 M AOE!- ; SET INVALID DISK ADDRESS STATUS
      04B3 1042 RM ER1 M_TAE,R2 ; DISK ADDRESS OVERFLOW OR,
      04B5 1043 BNEQ FUNCXT ; INVALID DISK ADDRESS ERROR?
      04B8 1044 MOVZWL #SS$ DRVERR,R0 ; IF NEQ YES
      04BA 1045 BITW #RM ER1 M DTE!- ; SET DRIVE ERROR STATUS
      04BF 1046 RM ER1 M_ILF!- ; DRIVE TIMING ERROR OR,
      04BF 1047 RM ER1 M_ILR!- ; ILLEGAL FUNCTION OR,
      04BF 1048 RM ER1 M_RMR!- ; ILLEGAL REGISTER OR,
      04BF 1049 RM ER1 M_WCF,R2 ; REGISTER MODIFY REFUSE OR,
      04C1 1050 BNEQ FUNCXT ; WRITE CLOCK FAIL ERROR?
      04C6 1051 MOVZWL #SS$ PARITY,R0 ; IF NEQ YES
      04CB 1052 BITW #RM ER1 M DCK!- ; Set parity error status.
      04CB 1053 RM ER1 M_ECH!- ; Data check error or,
      04CB 1054 RM ER1 M_HCRC,R2 ; ECC hard error or,
      04CD 1055 BNEQ FUNCXT ; header CRC error?
      04D3 1056 BBS #RM ER2 V BSE,UCBSW_DR_ER2(R5),FUNCXT ; Branch if so.
      04D8 1057 MOVZWL #SS$ CTRLERR,R0 ; IF SET, BAD SECTOR ERROR
      04DD 1058 BITW #RM ER1 M_HCE!- ; Set fatal controller error status.
      04DD 1059 BNEQ FUNCXT ; Header compare error or,
      04DF 1060 BITL #MBASH SR MAPPE!- ; parity error?
      04E6 1061 MBASH SR_MCPE!- ; Branch if so.
      04E6 1062 MBASH SR_SPE!- ; MAP PARITY ERROR OR,
      04E6 1063 MBASH SR_MDPE!- ; MASSBUS CONTROL PARITY ERROR OR,
      04E6 1064 MBASH SR_RDS,R1 ; MBA SILO PARITY ERROR OR,
      04E6 1065 BNEQ FUNCXT ; MASSBUS DATA PARITY ERROR OR,
      04E8 1066 MOVZWL #SS$ FORMAT,R0 ; READ DATA SUBSTITUTE?
      04ED 1067 BBS #RM ER1 V FER,R2,FUNCXT ; IF NEQ YES
      04F1 1068 MOVZWL #SS$ DATAHECK,R0 ; SET FORMAT ERROR STATUS
      04F6 1069 BITW #MBASH SR_WCKLWR!- ; IF SET, FORMAT ERROR
      04FB 1070 MBASH SR_WCKUPR,R1 ; SET DATA CHECK ERROR STATUS
      04FD 1071 BNEQ FUNCXT ; WRITE CHECK ERROR LOWER BYTE OR,
      0502 1072 MOVZWL #SS$ NONEXDRV,R0 ; WRITE CHECK ERROR UPPER BYTE?
      0502 1073 BBS #MBASH SR_NED,R1,FUNCXT ; IF NEQ YES
      0502 1074 ; SET NONEXISTENT DRIVE STATUS
      ; IF SET, NONEXISTENT DRIVE
```



```
50 0054 8F 3C 0506 1075      MOVZWL #SS$_CTRLERR,R0      ;SET CONTROLLER ERROR STATUS
      050B 1076
      050B 1077      :
      050B 1078      : FUNCTION COMPLETION COMMON EXIT
      050B 1079      :
      050B 1080
      050B 1081 FUNCXT:
      050B 1082      : FUNCTION EXIT
00000000 50 DD 050B 1082      PUSHL R0      ;SAVE FINAL REQUEST STATUS
      GF 16 050D 1083      JSB G*IOC$DIAGBUFILL      ;FILL DIAGNOSTIC BUFFER IF PRESENT
      0513 1084      RELCHAN      ;RELEASE CHANNEL IF OWNED
0092 C5 0A 91 0519 1085      CMPB #CDF_WRITECHECK,UCB$_FEX(R5) ;DRIVE RELATED FUNCTION?
      1A 1A 051E 1086      BGTRU 10$      ;IF GTRU YES
0092 C5 13 91 0520 1087      CMPB #CDF_READPRESET,UCB$_FEX(R5) ;DRIVE RELATED FUNCTION?
      13 1B 0525 1088      BLEQU 10$      ;IF LEQU YES
0092 C5 11 91 0527 1089      CMPB #CDF_AVAILABLE,UCB$_FEX(R5) ;DRIVE RELATED FUNCTION?
      0C 13 052C 1090      BEQL 10$      ;IF EQL YES
      52 58 A5 D0 052E 1091      MOVL UCB$_IRP(R5),R2      ;RETRIEVE ADDRESS OF IRP
      00D8 C5 A1 0532 1092      ADDW3 UCB$_DR_BCR(R5),-
02 AE 32 A2 0536 1093      IRP$_BCNT(R2),2(SP)      ; Calculate bytes transfered
      51 D4 053A 1094 10$: CLRL R1      ;CLEAR SECOND STATUS LONGWORD
      50 8ED0 053C 1095      POPL R0      ;RETRIEVE FINAL REQUEST STATUS
53 0091 C5 9A 053F 1096      MOVZBL UCB$_SLAVE+1(R5),R3      ;GET DRIVE OFFSET CONSTANT
53 0400 C443 DE 0544 1097      MOVAL MBAS$_ERB(R4)[R3],R3      ;GET ADDRESS OF DRIVE REGISTERS
      63 09 9A 054A 1098      MOVZBL #F_DRVCLR!1,RM_CS1(R3) ; Issue a drive clear before release.
      63 0B 9A 054D 1099      MOVZBL #F_RELEASE!1,RM_CS1(R3) ;RELEASE PORT
      0550 1100      REQCOM      ;COMPLETE REQUEST
```



0556 1102 .SBTTL HARDWARE FUNCTION EXECUTION  
0556 1103 :  
0556 1104 : FEX - HARDWARE FUNCTION EXECUTION  
0556 1105 :  
0556 1106 : THIS ROUTINE IS CALLED VIA A BSB WITH A BYTE IMMEDIATELY FOLLOWING THAT  
0556 1107 : SPECIFIES THE ADDRESS OF AN ERROR ROUTINE. ALL DATA IS ASSUMED TO HAVE BEEN  
0556 1108 : SET UP IN THE UCB BEFORE THE CALL. THE APPROPRIATE PARAMETERS ARE LOADED  
0556 1109 : INTO DEVICE REGISTERS AND THE FUNCTION IS INITIATED. IF THE FUNCTION IS AN  
0556 1110 : IMMEDIATE FUNCTION CONTROL RETURNS IMMEDIATELY. ELSE THE RETURN ADDRESS  
0556 1111 : IS STORED IN THE UCB AND A WAITFOR INTERRUPT IS EXECUTED. WHEN THE INTER-  
0556 1112 : RUPT OCCURS, CONTROL IS RETURNED TO THE CALLER.  
0556 1113 :  
0556 1114 : INPUTS:  
0556 1115 :  
0556 1116 : R0 = FUNCTION TABLE DISPATCH INDEX.  
0556 1117 : R3 = ADDRESS OF DRIVE CONTROL STATUS REGISTER 1.  
0556 1118 : R4 = ADDRESS OF MBA CONFIGURATION STATUS REGISTER.  
0556 1119 : R5 = DEVICE UNIT UCB ADDRESS.  
0556 1120 :  
0556 1121 : R0(SP) = RETURN ADDRESS OF CALLER.  
0556 1122 : R4(SP) = RETURN ADDRESS OF CALLER'S CALLER.  
0556 1123 :  
0556 1124 : IMMEDIATELY FOLLOWING INLINE AT THE CALL SITE IS A BYTE WHICH CONTAINS  
0556 1125 : A BRANCH DESTINATION TO AN ERROR RETRY ROUTINE.  
0556 1126 :  
0556 1127 : OUTPUTS:  
0556 1128 :  
0556 1129 : THERE ARE FOUR EXITS FROM THIS ROUTINE:  
0556 1130 :  
0556 1131 : 1. SPECIAL CONDITION - THIS EXIT IS TAKEN IF A POWER FAILURE OCCURS  
0556 1132 : OR THE OPERATION TIMES OUT. IT IS A JUMP TO THE APPROPRIATE  
0556 1133 : ERROR ROUTINE.  
0556 1134 :  
0556 1135 : 2. FATAL ERROR - THIS EXIT IS TAKEN IF A FATAL CONTROLLER OR DRIVE  
0556 1136 : ERROR OCCURS OR IF ANY ERROR OCCURS AND ERROR RETRY IS  
0556 1137 : INHIBITED. IT IS A JUMP TO THE FATAL ERROR EXIT ROUTINE.  
0556 1138 :  
0556 1139 : 3. RETRIABLE ERROR - THIS EXIT IS TAKEN IF A RETRIABLE CONTROLLER  
0556 1140 : OR DRIVE ERROR OCCURS AND ERROR RETRY IS NOT INHIBITED.  
0556 1141 : IT CONSISTS OF TAKING THE ERROR BRANCH EXIT.  
0556 1142 :  
0556 1143 : 4. SUCCESSFUL OPERATION - THIS EXIT IS TAKEN IF NO ERROR OCCURS  
0556 1144 : DURING THE OPERATION. IT CONSISTS OF A RETURN INLINE.  
0556 1145 :  
0556 1146 : IN ALL CASES IF AN ERROR OCCURS, AN ATTEMPT IS MADE TO LOG THE ERROR.  
0556 1147 :  
0556 1148 : IN ALL CASES FINAL DRIVE AND CONTROLLER REGISTERS ARE RETURNED VIA  
0556 1149 : THE GENERAL REGISTERS R0, R1, AND R2, AND THE UCB.  
0556 1150 :  
0556 1151 : R0 = DRIVE STATUS REGISTER.  
0556 1152 : R1 = MBA STATUS REGISTER.  
0556 1153 : R2 = DRIVE ERROR REGISTER 1.  
0556 1154 :  
0556 1155 : UCBSW\_EC1(R5) = ECC POSITION REGISTER.  
0556 1156 : UCBSW\_EC2(R5) = ECC PATTERN REGISTER.  
0556 1157 : UCBSW\_BCR(R5) = BYTE COUNT REGISTER.  
0556 1158 : UCBSW\_DR\_ER2(R5) = DRIVE ERROR REGISTER 2.



```
0093 C5 50 90 0091 C5 9A DE 0556 1159 :
38 A5 53 0400 C443 DF 0556 1160 :
50 0093 C5 9A 0556 1161 FEX:
0556 1162 :FUNCTION EXECUTOR
055B 1163 :SAVE DRIVER PC VALUE
0560 1164 :SAVE CASE INDEX
0565 1165 :GET DRIVE OFFSET CONSTANT
056B 1166 :GET ADDRESS OF DRIVE REGISTERS
0573 1167 :DUAL PORTED DRIVE?
0575 1168 GO: SEIZE :IF NEQ, YES
057A 1169 MOVZBL UCBSB_CEX(R5),R0 :Restore case index (func. code)
057A 1170 CASE R0,- :DISPATCH TO PROPER FUNCTION ROUTINE
057A 1171 POSIT,- :SEEK CYLINDER
057A 1172 EXFNC,- :RECALIBRATE
057A 1173 IMMED,- :DRIVE CLEAR
057A 1174 IMMED,- :RELEASE DRIVE
057A 1175 EXFNC,- :OFFSET HEADS
057A 1176 IMMED,- :RETURN TO CENTERLINE
057A 1177 POSIT,- :PACK ACKNOWLEDGE
057A 1178 XFER,- :SEARCH FOR SECTOR
057A 1179 XFER,- :WRITE CHECK
057A 1180 XFER,- :WRITE DATA
057A 1181 XFER,- :READ DATA
057A 1182 XFER,- :WRITE HEADER AND DATA
057A 1183 XFER,- :READ HEADER AND DATA
057A 1184 XFER,- :WRITE TRACK DESCRIPTOR
057A 1185 IMMED,- :READ TRACK DESCRIPTOR
057A 1186 XFER,- :AVAILABLE
057A 1187 IMMED,- :WRITE CHECK HEADER AND DATA
057A 1188 EXFNC,- :READIN PRESET
057A 1189 SEARCHA,- :DIAGNOSE
057A 1190 >,LIMIT=#CDF_SEEK :SEARCH AHEAD
05A6 1191 :
05A6 1192 : IMMEDIATE FUNCTION EXECUTION
05A6 1193 :
05A6 1194 : FUNCTIONS INCLUDE:
05A6 1195 :
05A6 1196 : NO OPERATION,
05A6 1197 : DRIVE CLEAR,
05A6 1198 : RELEASE PORT,
05A6 1199 : OFFSET,
05A6 1200 : READ IN PRESET, AND
05A6 1201 : PACK ACKNOWLEDGE.
05A6 1202 :
05A6 1203 : Two other functions which might (but hopefully don't) pass through this code
05A6 1204 : are UNLOAD and AVAILABLE. If such functions get here they are treated as
05A6 1205 : NOPs.
05A6 1206 :
05A6 1207 : THESE FUNCTIONS ARE EXECUTED IMMEDIATELY AND THE FINAL DEVICE REGISTERS
05A6 1208 : ARE RETURNED TO THE CALLER.
05A6 1209 :
05A6 1210 :
05A6 1211 : IMMEDIATE FUNCTION EXECUTION
05A6 1212 IMMED: DSBINT :DISABLE INTERRUPTS
05A6 1213 BBS #UCBSV_POWER,UCBSW_STS(R5),108 :IF SET, POWER HAS FAILED
09 64 A5 05 E0 05AC 1214 MOVZBL #F_DRVCLR!1,RM_CS1(R3) :CLEAR DRIVE ERRORS
63 09 9A 05B1 1215
```



```
63  FADF CF40 9A 05B4 1216      MOVZBL FTAB[R0],RM_CS1(R3)      ;EXECUTE FUNCTION
    010B 31 05BA 1217 10$: BRW ENBXIT      ;
    05BD 1218      ;
    05BD 1219      ;
    05BD 1220      ; ATTEMPT TO SEIZE THE PORT ON A DUAL PORTED DISK.
    05BD 1221      ;
    05BD 1222      ;
    B2 00D4 C5 03 E1 05BD 1223 SEIZE: BBC #DR_V_DUALPORT -      ; IF CLEAR, DUALPORT KIT
    05C3 1224      ; IS NOT PRESENT
51  00000064 8F D0 05C3 1225      MOVL UCB$B-DR_SSTS(R5),GO      ; Initialize count for the number of
    05CA 1226      ; times we will accept the loss of
    05CA 1227      ; the port while we are on the I/O
    05CA 1228      ; fork queue.
    05CA 1229 2$: DSBINT      ;DISABLE INTERRUPTS
    05D0 1230      CLRL RM_DS(R3)      ;ATTEMPT TO SEIZE PORT
    05D3 1231      BITL #RM_DS_M DPR,-      ;DID WE SEIZE THE PORT?
    05D9 1232      RM_DS(R3)
    05DB 1233      BNEQ 4$      ;IF NEQ, WE SEIZED THE PORT
    05DD 1234      WFIKPCW RETREG,#15      ;LET'S WAIT FOR THE PORT, ELSE TIMEOUT
    05E7 1235      IOFORK      ;CREATE FORK PROCESS
    DA 51 F4 05ED 1236      SOBGEQ R1,2$      ; Loop to make sure we really still
    05F0 1237      ; have the port after we are dequeued
    05F0 1238      ; off the I/O fork queue.
    00D8 31 05F0 1239      BRW RETREG      ; Otherwise, error - We keep losing the
    05F3 1240      ; port and we've retried enough.
    FF7C 31 05F3 1241 4$: ENBINT      ;ENABLE INTERRUPTS
    05F6 1242      BRW GO      ;LET'S CONTINUE, WE HAVE THE PORT
    05F9 1243      ;
    05F9 1244      ;
    05F9 1245      ; SEARCH AHEAD FUNCTION EXECUTION
    05F9 1246      ;
    05F9 1247      ; THIS FUNCTION MINIMIZES ROTATIONAL LATENCY BY SEARCHING FOR THE SECTOR THAT IS
    05F9 1248      ; FOUR SECTORS AHEAD OF THE STARTING SECTOR OF A TRANSFER.
    05F9 1249      ;
    05F9 1250      ; THE DESIRED CYLINDER, TRACK, AND SECTOR ADDRESS REGISTERS ARE LOADED, THE
    05F9 1251      ; FUNCTION IS INITIATED, AND A WAITFOR INTERRUPT IS EXECUTED. WHEN THE INTER-
    05F9 1252      ; RUPT OCCURS, THE FINAL DEVICE REGISTERS ARE RETURNED TO THE CALLER.
    05F9 1253      ;
    05F9 1254      ;
    51  00BC C5 3C 05F9 1255 SEARCHA:      ;
    51  51 04 82 05F9 1256      MOVZWL UCB$W_DA(R5),R1      ;GET DESIRED TRACK AND SECTOR ADDRESS
    04 18 05FE 1257      SUBB #4,R1      ;COMPUTE FOUR SECTORS BEFORE IT
    04 80 0601 1258      BGEQ 10$      ;IF GEQ BEFORE SECTOR ZERO
    51  44 A5 80 0603 1259      ADDB UCB$B_SECTORS(R5),R1      ;CONVERT TO AFTER SECTOR ZERO
    14 A3 51 D0 0607 1260 10$: MOVL R1,RM_DA(R3)      ;SET TRACK AND SECTOR ADDRESS
    15 11 060B 1261      BRB LDCYL      ;
    060D 1262      ;
    060D 1263      ; TRANSFER FUNCTION EXECUTION
    060D 1264      ;
    060D 1265      ;
    060D 1266      ; FUNCTIONS INCLUDE:
    060D 1267      ;
    060D 1268      ; WRITE TRACK DESCRIPTOR,
    060D 1269      ; WRITE CHECK,
    060D 1270      ; WRITE CHECK HEADER AND DATA,
    060D 1271      ; WRITE DATA,
    060D 1272      ; WRITE HEADER AND DATA,
```



```
060D 1273 : READ TRACK DESCRIPTOR,
060D 1274 : READ DATA, AND
060D 1275 : READ HEADER AND DATA.
060D 1276 :
060D 1277 : THE MAP REGISTERS, BYTE COUNT REGISTER, AND VIRTUAL ADDRESS REGISTER ARE
060D 1278 : LOADED FOLLOWED BY THE DESIRED CYLINDER, TRACK, AND SECTOR ADDRESS REGISTERS.
060D 1279 : THE FUNCTION IS INITIATED AND A WAITFOR INTERRUPT IS EXECUTED. WHEN THE
060D 1280 : INTERRUPT OCCURS, THE FINAL DEVICE REGISTERS ARE RETURNED TO THE CALLER.
060D 1281 :
060D 1282 : IT ASSUMED THAT THE CALLER OWNS THE CHANNEL ON WHICH THE I/O IS TO OCCUR.
060D 1283 :
060D 1284 :
060D 1285 XFER: ;TRANSFER FUNCTION EXECUTION
060D 1286 MCOML #0,MBASL_SR(R4) ;CLEAR MASSBUS ADAPTER ERRORS
0611 1287 LOADMBA ;LOAD MAP, BYTE COUNT, AND VIRTUAL ADDRESS
50 0093 C5 9A 0617 1288 MOVZBL UCBSB_CEX(R5),R0 ;RETRIEVE FUNCTION TABLE INDEX
061C 1289
061C 1290 :
061C 1291 : POSITIONING FUNCTION EXECUTION
061C 1292 :
061C 1293 : FUNCTIONS INCLUDE:
061C 1294 :
061C 1295 : SEEK CYLINDER, AND
061C 1296 : SEARCH FOR SECTOR.
061C 1297 :
061C 1298 : THE DESIRED CYLINDER, TRACK, AND SECTOR ADDRESS REGISTERS ARE LOADED, THE
061C 1299 : FUNCTION IS INITIATED, AND A WAITFOR INTERRUPT IS EXECUTED. WHEN THE INTER-
061C 1300 : RUPT OCCURS, THE FINAL DEVICE REGISTERS ARE RETURNED TO THE CALLER.
061C 1301 :
061C 1302 :
061C 1303 POSIT: ;POSITION FUNCTION EXECUTION
14 A3 00BC C5 3C 061C 1304 MOVZWL UCBSW_DA(R5),RM_DA(R3) ;SET DESIRED TRACK AND SECTOR ADDRESS
0622 1305 LDCYL: ;
28 A3 00BE C5 3C 0622 1306 MOVZWL UCBSW_DC(R5),RM_DC(R3) ;SET DESIRED CYLINDER ADDRESS
0628 1307
0628 1308 :
0628 1309 : INTERRUPT WAIT FUNCTION EXECUTION
0628 1310 :
0628 1311 : FUNCTIONS INCLUDE:
0628 1312 :
0628 1313 : DIAGNOSE,
0628 1314 : RECALIBRATE, AND
0628 1315 : RETURN TO CENTERLINE.
0628 1316 :
0628 1317 : THE OFFSET REGISTER IS LOADED, THE FUNCTION IS INITIATED, AND A WAITFOR
0628 1318 : INTERRUPT IS EXECUTED. WHEN THE INTERRUPT OCCURS, THE FINAL DEVICE REGISTERS
0628 1319 : ARE RETURNED TO THE CALLER.
0628 1320 :
0628 1321 :
0628 1322 EXFNC: ;EXECUTE FUNCTION
30 00D4 63 09 9A 0628 1323 MOVZBL #F DRVCLR!1,RM_CS1(R3) ;CLEAR DRIVE ERRORS
63 01 E1 0628 1324 BBC #DR V OM,UCBSB_DR SST5(R5),10$ ;IF CLR, NO OFFSET ACTIVE
63 0D 9A 0631 1325 MOVZBL #F OFFSET!1,RM_CST(R3) ;SET DRIVE IN OFFSET MODE
41 07 91 0634 1326 CMPB #DT$ RP07- ;Is this drive an RP07?
27 A5 12 0636 1327 UCBSB_DEVTYPE(R5)
0638 1328 BNEQ 10$ ; If not, no need to wait.
063A 1329 TIMEWAIT - ; Wait for 5 milliseconds or until
```



Address	Hex	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418
---------	-----	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------



```
06DD 1387 : value is that of the number of bytes transferred to memory (low word)
06DD 1388 : while on a write the more conservative value is the number of
06DD 1389 : bytes transferred to the drive (high word). Here we deposit
06DD 1390 : the entire register into a longword in the UCB. If the operation
06DD 1391 : was a read we leave the value as is. However if the operation
06DD 1392 : was a write (or anything but a read) we move the high word to
06DD 1393 : the low word in memory. All other pieces of this driver use the
06DD 1394 : low word of this longword as the valid byte count.
06DD 1395 :
10 A4 D0 06DD 1396 MOVL MBASL_BCR(R4),- ; Save entire byte count register
00D8 C5 06E0 1397 UCB$S_DR_BCR(R5) ; in the UCB.
50 58 A5 D0 06E3 1398 MOVL UCB$S_IRP(R5),R0 ; Retrieve IRP pointer.
01 E0 06E7 1399 BBS #IRP$V_FUNC,- ; If we had a read operation then
07 2A A0 06E9 1400 IRP$W_STS(R0),5$ ; just branch around since all OK.
00DA C5 B0 06EC 1401 MOVL UCB$S_DR_BCR+2(R5),- ; If NOT read, then copy high word to
00D8 C5 06F0 1402 UCB$S_DR_BCR(R5) ; low order word for later use.
00D6 C5 24 A3 F7 06F3 1403 5$: CVTLW RM_OF(R3),UCB$W_DR_OF(R5) ;SAVE OFFSET REGISTER
50 04 A3 D0 06F9 1405 MOVL RM_DS(R3),R0 ;GET CONTENTS OF DRIVE STATUS REGISTER
06FD 1406 ERROR:
51 00CC C5 D0 06FD 1407 MOVL UCB$S_DR_SR(R5),R1 ;RETRIEVE FINAL CONTROLLER STATUS
52 08 A3 D0 0702 1408 MOVL RM_ERT(R3),R2 ;GET CONTENTS OF DRIVE ERROR REGISTER 1
64 A5 0060 8F B3 0706 1409 BITW #UCB$M_POWER!,- ;POWER FAIL OR DEVICE TIMEOUT?
070C 1410 UCB$M_TIMEOUT,UCB$W_STS(R5) ;
03 13 070C 1411 BEQL 10$ ;IF EQL - NO
0137 31 070E 1412 BRW SPECOND ;BRANCH TO SPECIAL CONDITION
0093 C5 15 91 0711 1413 10$: CMPB #CDF_SEARCHA,UCB$B_CEX(R5) ;Search Ahead?
5F 13 0716 1414 BEQL SAFUNC ;Branch to special search-ahead code.
0093 C5 0A 91 0718 1415 CMPB #CDF_WRITECHECK,UCB$B_CEX(R5) ;DRIVE RELATED FUNCTION?
07 1A 071D 1416 BGTRU 20$ ;IF GTRU - YES
0093 C5 13 91 071F 1417 CMPB #CDF_READPRESET,UCB$B_CEX(R5) ;DRIVE RELATED FUNCTION?
03 1A 0724 1418 BGTRU 30$ ;IF GTRU - NO
0083 31 0726 1419 20$: BRW DFUNC ;DRIVE FUNCTION
0729 1420
0729 1421 :
0729 1422 : CHECK FOR RM80 SKIP SECTOR ERROR
0729 1423 :
0729 1424 :
41 A5 0D 91 0729 1425 30$: CMPB #DT$ RM80,UCB$B_DEVTYPE(R5) ;RM80?
56 12 072D 1426 BNEQ CFUNC ;IF NEQ - NO
05 E1 072F 1427 BBC #RM_ER2_V_SSE,- ; If clear, No Skip Sectoring error,
50 00D0 C5 0731 1428 UCB$W_DR_ER2(R5),CFUNC ; so branch around.
52 0180 8F B3 0735 1429 BITW #RM_ER1_M_HCE!RM_ER1_M_HCRC,R2 ;HEADER COMPARE OR HDR CRC ERR?
49 12 073A 1430 BNEQ CFUNC ;IF NEQ - YES
00C9 C5 02 88 073C 1431 BISB #RM_OF_M_SSE1/256,UCB$W_OFFSET+1(R5) ;SET SKIP SECTOR INHIBIT
50 50 DD 0741 1432 RO ;SAVE R0 (DRIVE STATUS REGISTER)
00D8 C5 3C 0743 1433 MOVZWL UCB$S_DR_BCR(R5),R0 ; Get negative bytes remaining
05 12 0748 1434 BNEQ 40$ ;IF NEQ - PARTIAL TRANSFER
50 FFFF 8F B0 074A 1435 MOVW #-1,R0 ;FAKE A PARTIAL TRANSFER
50 7E A5 A0 074F 1436 40$: ADDW UCB$W_BCNT(R5),R0 ;CALCULATE BYTES TRANSFERRED
50 01FF 8F AA 0753 1437 BICW #X1FF,R0 ;TRUNCATE TO LAST BLOCK TRANSFERRED
00000000 GF 16 0758 1438 JSB G*IOCSUPDATRANSF ;UPDATE TRANSFER PARAMETERS
00BC C5 96 075E 1439 INCB UCB$W_DA(R5) ;INCREMENT TO NEXT SECTOR
50 8ED0 0762 1440 POPL R0 ;RESTORE R0 (DRIVE STATUS REGISTER)
03 00D4 C5 00 E0 0765 1441 BBS #DR_V_DCK,UCB$B_DR_SSTS(R5),45$ ;IF SET - DATACHECK IN PROGRESS
FB22 31 076B 1442 BRW TRANNOCH ;RESTART TRANSFER
076E 1443 45$: RELCHAN ;RELEASE CHANNEL
```



```
FB70 31 0774 1444 BRW CHECKRETRY ;RESTART DATA CHECK
      0777 1445 ;NO SSE - CHECK FOR TRACK-TRACK SSEI CLR
      0777 1446 .ENABL LSB
      0777 1447
      0777 1448
      0777 1449 : SEARCH AHEAD ERROR CHECKING
      0777 1450 :
      0777 1451
      0777 1452 SAFUNC:
7A 00D0 C5 0E E1 0777 1453 BBC #RM_ER2_V_SKI, - ;The only error worth checking on
      077D 1454 UCB$W_DR_ER2(R5), 30$ ;search-ahead is seek incomplete.
      00000000'GF 16 077D 1455 JSB G^ERL$DEVICERR ;SKI errors, however, must be logged
      68 11 0783 1456 BRB 25$ ;and retried.
      0785 1457
      0785 1458 : CONTROLLER RELATED FUNCTION
      0785 1459 :
      0785 1460 :
      0785 1461
      0785 1462 CFUNC:
51 000E5FFF 8F D3 0785 1463 BITL #MBASH_ERROR,R1 ;ANY CONTROLLER ERRORS?
      69 13 078C 1464 BEQL 30$ ;IF EQL NO
      00000000'GF 16 078E 1465 JSB G^ERL$DEVICERR ;ALLOCATE AND FILL ERROR MESSAGE BUFFER
66 009A C5 0F E0 0794 1466 BBS #IOSV_INHRETRY,UCB$W_FUNC(R5),40$ ;IF SET, RETRY INHIBITED
51 0008000B 8F D3 079A 1467 BITL #MBASH_SR_ERCONF!- ;ERROR CONFIRMATION OR,
      07A1 1468 MBASH_SR_ISTO!- ;INTERFACE SEQUENCE TIMEOUT OR,
      07A1 1469 MBASH_SR_PGE!- ;PROGRAMMING ERROR OR,
      07A1 1470 MBASH_SR_RDTO,R1 ;READ TIMEOUT?
      07A1 1471 BNEQ 40$ ;IF NEQ YES - FATAL CONTROLLER ERROR
51 00064FF4 8F D3 07A3 1472 BITL #MBASH_SR_DLT!- ;DATA LATE OR,
      07AA 1473 MBASH_SR_INVMAP!- ;INVALID MAP REGISTER OR,
      07AA 1474 MBASH_SR_MAPPE!- ;MAP REGISTER PARITY ERROR OR,
      07AA 1475 MBASH_SR_MBEXC!- ;MASSBUS EXCEPTION OR,
      07AA 1476 MBASH_SR_MCPE!- ;MASSBUS CONTROL PARITY ERROR OR,
      07AA 1477 MBASH_SR_SPE!- ;MBA SILO PARITY ERROR OR,
      07AA 1478 MBASH_SR_MDPE!- ;MASSBUS DATA PARITY ERROR OR,
      07AA 1479 MBASH_SR_MXF!- ;MISSED TRANSFER OR,
      07AA 1480 MBASH_SR_NED!- ;NONEXISTENT DRIVE OR,
      07AA 1481 MBASH_SR_RDS!- ;READ DATA SUBSTITUTE OR,
      07AA 1482 MBASH_SR_WCKLWR!- ;WRITE CHECK LOWER BYTE OR,
      07AA 1483 MBASH_SR_WCKUPR,R1 ;WRITE CHECK UPPER BYTE?
      1B 12 07AA 1484 BNEQ 20$ ;IF NEQ YES - RETRIABLE CONTROLLER ERROR
      07AC 1485
      07AC 1486 : DRIVE RELATED FUNCTION
      07AC 1487 :
      07AC 1488 :
      07AC 1489
      07AC 1490 DFUNC:
      47 50 0E E1 07AC 1491 10$: BBC #RM_DS_V_ERR,R0,30$ ;IF CLR, NO DRIVE ERRORS
      7E A5 AE 07B0 1492 MNEGW UCB$W_BCNT(R5) - ;Reset byte count - NO TRANSFER
      00D8 C5 07B3 1493 UCB$W_DR_BCR(R5) ; Don't log error if Medium offline at
      45 00D5 C5 E8 07B6 1494 BLBS UCB$W_DR_ERL(R5),40$ ; start of function.
      00000000'GF 16 07BB 1495 JSB G^ERL$DEVICERR ;ALLOCATE AND FILL ERROR MESSAGE BUFFER
39 009A C5 0F E0 07C1 1497 BBS #IOSV_INHRETRY,UCB$W_FUNC(R5),40$ ;IF SET, RETRY INHIBITED
      35 50 0C E1 07C7 1498 20$: BBC #RM_DS_V_MOL,R0,40$ ;IF CLR, MEDIUM OFFLINE
      31 50 06 E1 07CB 1499 BBC #RM_DS_V_VV,R0,40$ ;IF CLR, INVALID VOLUME
      52 0180 8F B3 07CF 1500 BITW #RM_ERT_M_HCRC!- ;Check HCRC and HCE before checking
```



```

52 0E17 17 12 07D4 1501          RM_ER1_M_HCE,R2      ; BSE and FER.
                                25$                     ; NEQ means HCRC or HCE is set.
                                BNEQ                     ; ADDRESS OVERFLOW OR,
                                BITW                     ; FORMAT ERROR OR,
                                RM_ER1_M_AOE!-           ; INVALID ADDRESS OR,
                                RM_ER1_M_FER!-           ; ILLEGAL FUNCTION OR,
                                RM_ER1_M_IAE!-           ; ILLEGAL REGISTER OR,
                                RM_ER1_M_ILF!-           ; REGISTER MODIFY REFUSE OR,
                                RM_ER1_M_ILR!-           ; WRITE LOCK ERROR?
                                RM_ER1_M_RMR!-           ; IF NEQ YES - FATAL DRIVE ERROR
                                RM_ER1_M_WLE,R2         ; BAD SECTOR ERROR OR,
                                40$                     ; OPERATOR PLUG ERROR?
                                BITW                     ; IF NEQ YES - FATAL DRIVE ERROR
                                RM_ER2_M_OPE,UCBSW_DR_ER2(R5) ; Is the drive unsafe?
                                BNEQ                     ; Branch if so.
                                45$
                                07DB 1502
                                07DB 1503
                                07DB 1504
                                07DB 1505
                                07DB 1506
                                07DB 1507
                                07DB 1508
                                07DB 1509
                                07DB 1510
                                07DD 1511
                                07E4 1512
                                07E4 1513
                                07E6 1514
                                07EB 1515
                                07ED 1516
                                07ED 1517
                                07ED 1518
                                07ED 1519
                                07ED 1520
                                07ED 1521 25$: CVTWL @UCBSL_DPC(R5),-(SP) ; GET BRANCH DISPLACEMENT
                                07F2 1522 ADDL (SP)+,UCBSL_DPC(R5) ; CALCULATE RETURN ADDRESS - 2
                                07F7 1523 30$: ADDL #2,UCBSL_DPC(R5) ; SKIP PAST BRANCH DISPLACEMENT WORD
                                07FC 1524 JMP @UCBSL_DPC(R5) ; RETURN TO DRIVER
                                0800 1525
                                0800 1526
                                0800 1527
                                0800 1528
                                0800 1529
                                FC70 31 0800 1530 40$: BRW FATALERR ;
                                0803 1531
                                0803 1532
                                0803 1533
                                0803 1534
                                0803 1535
                                0803 1536 45$: DSBINT ; Disable interrupts.
                                0809 1537 BBC #UCBSV POWER,- ; Branch if no power failure occurred.
                                080B 1538 UCBSW_STS(R5),47$ ;
                                080E 1539 BRW ENBXIT ; Otherwise, enable interrupts and
                                0811 1540 ; go process error.
                                0811 1541 47$: MOVZBL #F_DRVCLR!1,RM_CS1(R3) ; Attempt to clear unsafe condition.
                                0814 1542 TIMEWAIT - ; Wait for ten microseconds or until
                                0814 1543 TIME = #1,- ; unsafe condition clears.
                                0814 1544 BITVAL = #RM_ER1_M_UNSAFE,- ;
                                0814 1545 SOURCE = RM_ER1(R3),- ;
                                0814 1546 CONTEXT = L,- ;
                                0814 1547 SENSE = .FALSE. ;
                                083C 1548 ENBINT ; Enable interrupts.
                                083F 1549 MOVL RM_ER1(R3),R2 ; Retrieve error status.
                                0843 1550 BLBS R0,25$ ; Branch if drive is no longer unsafe.
                                0846 1551 BRB 40$ ; Otherwise, fatal error.
                                0848 1552
                                0848 1553
                                0848 1554
                                0848 1555
                                0848 1556
                                0848 1557
                                0848 1558
                                0848 1559
                                0848 1560
                                0848 1561
                                0848 1562
                                0848 1563
                                0848 1564
                                0848 1565
                                0848 1566
                                0848 1567
                                0848 1568
                                0848 1569
                                0848 1570
                                0848 1571
                                0848 1572
                                0848 1573
                                0848 1574
                                0848 1575
                                0848 1576
                                0848 1577
                                0848 1578
                                0848 1579
                                0848 1580
                                0848 1581
                                0848 1582
                                0848 1583
                                0848 1584
                                0848 1585
                                0848 1586
                                0848 1587
                                0848 1588
                                0848 1589
                                0848 1590
                                0848 1591
                                0848 1592
                                0848 1593
                                0848 1594
                                0848 1595
                                0848 1596
                                0848 1597
                                0848 1598
                                0848 1599
                                0848 1600
                                0848 1601
                                0848 1602
                                0848 1603
                                0848 1604
                                0848 1605
                                0848 1606
                                0848 1607
                                0848 1608
                                0848 1609
                                0848 1610
                                0848 1611
                                0848 1612
                                0848 1613
                                0848 1614
                                0848 1615
                                0848 1616
                                0848 1617
                                0848 1618
                                0848 1619
                                0848 1620
                                0848 1621
                                0848 1622
                                0848 1623
                                0848 1624
                                0848 1625
                                0848 1626
                                0848 1627
                                0848 1628
                                0848 1629
                                0848 1630
                                0848 1631
                                0848 1632
                                0848 1633
                                0848 1634
                                0848 1635
                                0848 1636
                                0848 1637
                                0848 1638
                                0848 1639
                                0848 1640
                                0848 1641
                                0848 1642
                                0848 1643
                                0848 1644
                                0848 1645
                                0848 1646
                                0848 1647
                                0848 1648
                                0848 1649
                                0848 1650
                                0848 1651
                                0848 1652
                                0848 1653
                                0848 1654
                                0848 1655
                                0848 1656
                                0848 1657
                                0848 1658
                                0848 1659
                                0848 1660
                                0848 1661
                                0848 1662
                                0848 1663
                                0848 1664
                                0848 1665
                                0848 1666
                                0848 1667
                                0848 1668
                                0848 1669
                                0848 1670
                                0848 1671
                                0848 1672
                                0848 1673
                                0848 1674
                                0848 1675
                                0848 1676
                                0848 1677
                                0848 1678
                                0848 1679
                                0848 1680
                                0848 1681
                                0848 1682
                                0848 1683
                                0848 1684
                                0848 1685
                                0848 1686
                                0848 1687
                                0848 1688
                                0848 1689
                                0848 1690
                                0848 1691
                                0848 1692
                                0848 1693
                                0848 1694
                                0848 1695
                                0848 1696
                                0848 1697
                                0848 1698
                                0848 1699
                                0848 1700
                                0848 1701
                                0848 1702
                                0848 1703
                                0848 1704
                                0848 1705
                                0848 1706
                                0848 1707
                                0848 1708
                                0848 1709
                                0848 1710
                                0848 1711
                                0848 1712
                                0848 1713
                                0848 1714
                                0848 1715
                                0848 1716
                                0848 1717
                                0848 1718
                                0848 1719
                                0848 1720
                                0848 1721
                                0848 1722
                                0848 1723
                                0848 1724
                                0848 1725
                                0848 1726
                                0848 1727
                                0848 1728
                                0848 1729
                                0848 1730
                                0848 1731
                                0848 1732
                                0848 1733
                                0848 1734
                                0848 1735
                                0848 1736
                                0848 1737
                                0848 1738
                                0848 1739
                                0848 1740
                                0848 1741
                                0848 1742
                                0848 1743
                                0848 1744
                                0848 1745
                                0848 1746
                                0848 1747
                                0848 1748
                                0848 1749
                                0848 1750
                                0848 1751
                                0848 1752
                                0848 1753
                                0848 1754
                                0848 1755
                                0848 1756
                                0848 1757
                                0848 1758
                                0848 1759
                                0848 1760
                                0848 1761
                                0848 1762
                                0848 1763
                                0848 1764
                                0848 1765
                                0848 1766
                                0848 1767
                                0848 1768
                                0848 1769
                                0848 1770
                                0848 1771
                                0848 1772
                                0848 1773
                                0848 1774
                                0848 1775
                                0848 1776
                                0848 1777
                                0848 1778
                                0848 1779
                                0848 1780
                                0848 1781
                                0848 1782
                                0848 1783
                                0848 1784
                                0848 1785
                                0848 1786
                                0848 1787
                                0848 1788
                                0848 1789
                                0848 1790
                                0848 1791
                                0848 1792
                                0848 1793
                                0848 1794
                                0848 1795
                                0848 1796
                                0848 1797
                                0848 1798
                                0848 1799
                                0848 1800
                                0848 1801
                                0848 1802
                                0848 1803
                                0848 1804
                                0848 1805
                                0848 1806
                                0848 1807
                                0848 1808
                                0848 1809
                                0848 1810
                                0848 1811
                                0848 1812
                                0848 1813
                                0848 1814
                                0848 1815
                                0848 1816
                                0848 1817
                                0848 1818
                                0848 1819
                                0848 1820
                                0848 1821
                                0848 1822
                                0848 1823
                                0848 1824
                                0848 1825
                                0848 1826
                                0848 1827
                                0848 1828
                                0848 1829
                                0848 1830
                                0848 1831
                                0848 1832
                                0848 1833
                                0848 1834
                                0848 1835
                                0848 1836
                                0848 1837
                                0848 1838
                                0848 1839
                                0848 1840
                                0848 1841
                                0848 1842
                                0848 1843
                                0848 1844
                                0848 1845
                                0848 1846
                                0848 1847
                                0848 1848
                                0848 1849
                                0848 1850
                                0848 1851
                                0848 1852
                                0848 1853
                                0848 1854
                                0848 1855
                                0848 1856
                                0848 1857
                                0848 1858
                                0848 1859
                                0848 1860
                                0848 1861
                                0848 1862
                                0848 1863
                                0848 1864
                                0848 1865
                                0848 1866
                                0848 1867
                                0848 1868
                                0848 1869
                                0848 1870
                                0848 1871
                                0848 1872
                                0848 1873
                                0848 1874
                                0848 1875
                                0848 1876
                                0848 1877
                                0848 1878
                                0848 1879
                                0848 1880
                                0848 1881
                                0848 1882
                                0848 1883
                                0848 1884
                                0848 1885
                                0848 1886
                                0848 1887
                                0848 1888
                                0848 1889
                                0848 1890
                                0848 1891
                                0848 1892
                                0848 1893
                                0848 1894
                                0848 1895
                                0848 1896
                                0848 1897
                                0848 1898
                                0848 1899
                                0848 1900
                                0848 1901
                                0848 1902
                                0848 1903
                                0848 1904
                                0848 1905
                                0848 1906
                                0848 1907
                                0848 1908
                                0848 1909
                                0848 1910
                                0848 1911
                                0848 1912
                                0848 1913
                                0848 1914
                                0848 1915
                                0848 1916
                                0848 1917
                                0848 1918
                                0848 1919
                                0848 1920
                                0848 1921
                                0848 1922
                                0848 1923
                                0848 1924
                                0848 1925
                                0848 1926
                                0848 1927
                                0848 1928
                                0848 1929
                                0848 1930
                                0848 1931
                                0848 1932
                                0848 1933
                                0848 1934
                                0848 1935
                                0848 1936
                                0848 1937
                                0848 1938
                                0848 1939
                                0848 1940
                                0848 1941
                                0848 1942
                                0848 1943
                                0848 1944
                                0848 1945
                                0848 1946
                                0848 1947
                                0848 1948
                                0848 1949
                                0848 1950
                                0848 1951
                                0848 1952
                                0848 1953
                                0848 1954
                                0848 1955
                                0848 1956
                                0848 1957
                                0848 1958
                                0848 1959
                                0848 1960
                                0848 1961
                                0848 1962
                                0848 1963
                                0848 1964
                                0848 1965
                                0848 1966
                                0848 1967
                                0848 1968
                                0848 1969
                                0848 1970
                                0848 1971
                                0848 1972
                                0848 1973
                                0848 1974
                                0848 1975
                                0848 1976
                                0848 1977
                                0848 1978
                                0848 1979
                                0848 1980
                                0848 1981
                                0848 1982
                                0848 1983
                                0848 1984
                                0848 1985
                                0848 1986
                                0848 1987
                                0848 1988
                                0848 1989
                                0848 1990
                                0848 1991
                                0848 1992
                                0848 1993
                                0848 1994
                                0848 1995
                                0848 1996
                                0848 1997
                                0848 1998
                                0848 1999
                                0848 2000
                                0848 2001
                                0848 2002
                                0848 2003
                                0848 2004
                                0848 2005
                                0848 2006
                                0848 2007
                                0848 2008
                                0848 2009
                                0848 2010
                                0848 2011
                                0848 2012
                                0848 2013
                                0848 2014
                                0848 2015
                                0848 2016
                                0848 2017
                                0848 2018
                                0848 2019
                                0848 2020
                                0848 2021
                                0848 2022
                                0848 2023
                                0848 2024
                                0848 2025
                                0848 2026
                                0848 2027
                                0848 2028
                                0848 2029
                                0848 2030
                                0848 2031
                                0848 2032
                                0848 2033
                                0848 2034
                                0848 2035
                                0848 2036
                                0848 2037
                                0848 2038
                                0848 2039
                                0848 2040
                                0848 2041
                                0848 2042
                                0848 2043
                                0848 2044
                                0848 2045
                                0848 2046
                                0848 2047
                                0848 2048
                                0848 2049
                                0848 2050
                                0848 2051
                                0848 2052
                                0848 2053
                                0848 2054
                                0848 2055
                                0848 2056
                                0848 2057
                                0848 2058
                                0848 2059
                                0848 2060
                                0848 2061
                                0848 2062
                                0848 2063
                                0848 2064
                                0848 2065
                                0848 2066
                                0848 2067
                                0848 2068
                                0848 2069
                                0848 2070
                                0848 2071
                                0848 2072
                                0848 2073
                                0848 2074
                                0848 2075
                                0848 2076
                                0848 2077
                                0848 2078
                                0848 2079
                                0848 2080
                                0848 2081
                                0848 2082
                                0848 2083
                                0848 2084
                                0848 2085
                                0848 2086
                                0848 2087
                                0848 2088
                                0848 2089
                                0848 2090
                                0848 2091
                                0848 2092
                                0848 2093
                                0848 2094
                                0848 2095
                                0848 2096
                                0848 2097
                                0848 2098
                                0848 2099
                                0848 2100
                                0848 2101
                                0848 2102
                                0848 2103
                                0848 2104
                                0848 2105
                                0848 2106
                                0848 2107
                                0848 2108
                                0848 2109
                                0848 2110
                                0848 2111
                                0848 2112
                                0848 2113
                                0848 2114
                                0848 2115
                                0848 2116
                                0848 2117
                                0848 2118
                                0848 2119
                                0848 2120
                                0848 2121
                                0848 2122
                                0848 2123
                                0848 2124
                                0848 2125
                                0848 2126
                                0848 2127
                                0848 2128
                                0848 2129
                                0848 2130
                                0848 2131
                                0848 2132
                                0848 2133
                                0848 2134
                                0848 2135
                                0848 2136
                                0848 2137
                                0848 2138
                                0848 2139
                                0848 2140
                                0848 2141
                                0848 2142
                                0848 2143
                                0848 2144
                                0848 2145
                                0848 2146
                                0848 2147
                                0848 2148
                                0848 2149
                                0848 2150
                                0848 2151
                                0848 2152
                                0848 2153
                                0848 2154
                                0848 2155
                                0848 2156
                                0848 2157
                                0848 2158
                                0848 2159
                                0848 2160
                                0848 2161
                                0848 2162
                                0848 2163
                                0848 2164
                                0848 2165
                                0848 2166
                                0848 2167
                                0848 2168
                                0848 2169
                                0848 2170
                                0848 2171
                                0848 2172
                                0848 2173
                                0848 2174
                                0848 2175
                                0848 2176
                                0848 2177
                                0848 2178
                                0848 2179
                                0848 2180
                                0848 2181
                                0848 2182
                                0848 2183
                                0848 2184
                                0848 2185
                                0848 2186
                                0848 2187
                                0848 2188
                                0848 2189
                                0848 2190
                                0848 2191
                                0848 2192
                                0848 2193
                                0848 2194
                                0848 2195
                                0848 2196
                                0848 2197
                                0848 2198
                                0848 2199
                                0848 2200
                                0848 2201
                                0848 2202
                                0848 2203
                                0848 2204
                                0848 2205
                                0848 2206
                                0848 2207
                                0848 2208
                                0848 2209
                                0848 2210
                                0848 2211
                                0848 2212
                                0848 2213
                                0848 2214
                                0848 2215
                                0848 2216
                                0848 2217
                                0848 2218
                                0848 2219
                                0848 2220
                                0848 2221
                                0848 2222
                                0848 2223
                                0848 2224
                                0848 2225
                                0848 2226
                                0848 2227
                                0848 2228
                                0848 2229
                                0848 2230
                                0848 2231
                                0848 2232
                                0848 2233
                                0848 2234
                                0848 2235
                                0848 2236
                                0848 2237
                                0848 2238
                                0848 2239
                                0848 2240
                                0848 2241
                                0848 2242
                                0848 2243
                                0848 2244
                                0848 2245
                                0848 2246
                                0848 2247
                                0848 2248
                                0848 2249
                                0848 2250
                                0848 2251
                                0848 2252
                                0848 2253
                                0848 2254
                                0848 2255
                                0848 2256
                                0848 2257
                                0848 2258
                                0848 2259
                                0848 2260
                                0848 2261
                                0848 2262
                                0848 2263
                                0848 2264
                                0848 2265
                                0848 2266
                                0848 2267
                                0848 2268
                                0848 2269
                                0848 2270
                                0848 2271
                                0848 2272
                                0848 2273
                                0848 2274
                                0848 2275
                                0848 2276
                                0848 2277
                                0848 2278
                                0848 2279
                                0848 2280
                                0848 2281
                                0848 2282
                                0848 2283
                                0848 2284
                                0848 2285
                                0848 2286
                                0848 2287
                                0848 2288
                                0848 2289
                                0848 2290
                                0848 2291
                                0848 2292
                                0848 2293
                                0848 2294
                                0848 2295
                                0848 2296
                                0848 2297
                                0848 2298
                                0848 2299
                                0848 2300
                                0848 2301
                                0848 2302
                                0848 2303
                                0848 2304
                                0848 2305
                                0848 2306
                                0848 2307
                                0848 2308
                                0848 2309
                                0848 2310
                                0848 2311
                                0848 2312
                                0848 2313
                                0848 2314
                                0848 2315
                                0848 2316
                                0848 2317
                                0848 2318
                                0848 2319
                                0848 2320
                                0848 2321
                                0848 2322
                                0848 2323
                                0848 2324
                                0848 2325
                                0848 2326
                                0848 2327
                                0848 2328
                                0848 2329
                                0848 2330
                                0848 2331
                                0848 2332
                                0848 2333
                                0848 2334
                                0848 2335
                                0848 2336
                                0848 2337
                                0848 2338
                                0848 2339
                                0848 2340
                                0848 2341
                                0848 2342
                                0848 2343
                                0848 2344
                                0848 2345
                                0848 2346
                                0848 2347
                                0848 2348
                                0848 2349
                                0848 2350
                                0848 2351
                                0848 2352
                                0848 2353
                                0848 2354
                                0848 2355
                                0848 2356
                                0848 2357
                                0848 2358
                                0848 2359
                                0848 2360
                                0848 2361
                                0848 2362
                                0848 2363
                                0848 2364
                                0848 2365
                                0848 2366
                                0848 2367
                                0848 2368
                                0848 2369
                                0848 2370
                                0848 2371
                                0848 2372
                                0848 2373
                                0848 2374
                                0848 2375
                                0848 2376
                                0848 2377
                                0848 2378
                                0848 2379
                                0848 2380
                                0848 2381
                                0848 2382
                                0848 2383
                                0848 2384
                                0848 2385
                                0848 2386
                                0848 2387
                                0848 2388
                                0848 2389
                                0848 2390
                                0848 2391
                                0848 2392
                                0848 2393
                                0848 2394
                                0848 2395
                                0848 2396
                                0848 2397
                                0848 2398
                                0848 2399
                                0848 2400
                                0848 2401
                                0848 2402
                                0848 2403
                                0848 2404
                                0848 2405
                                0848 2406
                                0848 2407
                                0848 2408
                                0848 2409
                                0848 2410
                                0848 2411
                                0848 2412
                                0848 2413
                                0848 2414
                                0848 2415
                                0848 2416
                                0848 2417
                                0848 2418
                                0848 2419
                                0848 2420
                                0848 2421
                                0848 2422
                                0848 2423
                                0848 2424
                                0848 2425
                                0848 2426
                                0848 2427
                                0848 2428
                                0848 2429
                                0848 2430
                                0848 2431
                                0848 2432
                                0848 2433
                                0848 2434
                                0848 2435
                                0848 2436
                                0848 2437
                                0848 2438
                                0848 2439
                                0848 2440
                                0848 2441
                                0848 2442
                                0848 2443
                                0848 2444
                                0848 2445
                                0848 2446
                                0848 2447
                                0848 2448
                                0848 2449
                                0848 2450
                                0848 2451
                                0848 2452
                                0848 2453
                                0848 2454
                                0848 2455
                                0848 2456
                                0848 2457
                                0848 2458
                                0848 2459
                                0848 2460
                                0848 2461
                                0848 2462
                                0848 2463
                                0848 2464
```



```
61 64 A5 05 E4 0848 1558 50$: BBSC #UCBSV_POWER,UCBSW_STS(R5),70$ ;IF SET, POWER FAILURE
      084D 1559
      084D 1560 :
      084D 1561 : DEVICE TIME OUT
      084D 1562 :
      084D 1563 :
00000000'GF 16 084D 1564 JSB G*ERL$DEVICTMO ;LOG DEVICE TIME OUT
53 24 A5 D0 0853 1565 MOVL UCBSL_CRB(R5),R3 ;GET ADDRESS OF CRB
53 2C A3 D0 0857 1566 MOVL CRBSL_INTD+VE($L_IDB(R3),R3 ;GET ADDRESS OF IDB
04 A3 55 D1 085B 1567 CML R5,IDBSL_OWNER(R3) ;DEVICE OWN CONTROLLER?
      22 12 085F 1568 BNEQ 60$ ;IF NEQ NO
      0861 1569 DSBINT ;DISABLE INTERRUPTS
      06 D0 0867 1570 MOVL #MBASM_CR_ABORT!MBASM_CR_IE,- ;ABORT THE DATA TRANSFER
04 A4 0869 1571 MBASL_CR(R4)
      086B 1572 WFIKPC 55$,#T5 ;WAIT FOR ABORT AND KEEP CHANNEL
      0875 1573 IOFORK ;CREATE FORK PROCESS
      087B 1574 55$:
04 A4 01 D0 087B 1575 MOVL #MBASM_CR_INIT,MBASL_CR(R4) ;INITIALIZE ENTIRE MBA
04 A4 04 D0 087F 1576 MOVL #MBASM_CR_IE,MBASL_CR(R4) ;ENABLE DEVICE INTERRUPTS
50 022C 8F 3C 0883 1577 60$: SETIPL UCBSB_FIPC(R5) ;LOWER TO FORK LEVEL
      0080 C5 97 0887 1578 MOVZWL #SS$ TIMEOUT,R0 ;SET DEVICE TIMEOUT STATUS
      OF 13 088C 1579 DECB UCBSB_ERTCNT(R5) ;ANY ERROR RETRIES REMAINING?
      0890 1580 BEQL RESETXFR ;IF EQL NO
      0892 1581 RELCHAN ;RELEASE CHANNEL IF OWNED
64 A5 0040 8F AA 0898 1582 BICW #UCBSM_TIMEOUT,UCBSW_STS(R5) ;CLEAR TIME OUT STATUS
      F919 31 089E 1583 BRW FDISPATCH
      08A1 1584
      08A1 1585 :
      08A1 1586 : RESET TRANSFER BYTE COUNT TO ZERO
      08A1 1587 :
      08A1 1588 :
      08A1 1589 RESETXFR:
53 58 A5 D0 08A1 1590 MOVL UCBSL_IRP(R5),R3 ;RETRIEVE ADDRESS OF I/O PACKET
      32 A3 AE 08A5 1591 MNEGW IRPSW_BCNT(R3),-
      00D8 C5 31 08AB 1592 BRW UCBSL_DR_BCR(R5) ; Reset transfer byte count
      FC5D 08AB 1593
      08AE 1594
      08AE 1595 :
      08AE 1596 : POWER FAILURE
      08AE 1597 :
      08AE 1598 :
      08AE 1599 70$: RELCHAN ;RELEASE CHANNEL
53 58 A5 D0 08B4 1600 MOVL UCBSL_IRP(R5),R3 ;RETRIEVE ADDRESS OF I/O PACKET
78 A5 2C A3 7D 08B8 1601 MOVQ IRPSL_SVAPTE(R3),UCBSL_SVAPTE(R5) ;RESTORE TRANSFER PARAMETERS
      F886 31 08BD 1602 BRW DR_STARTIO
      08C0 1603 .DSABL L$B
      08C0 1604
```



```
08C0 1606 .SBTTL REGISTER DUMP ROUTINE
08C0 1607 :
08C0 1608 : DR_REGDUMP - REGISTER DUMP ROUTINE
08C0 1609 :
08C0 1610 : THIS ROUTINE IS CALLED TO SAVE THE CONTROLLER AND DRIVE REGISTERS IN A
08C0 1611 : SPECIFIED BUFFER. IT IS CALLED FROM THE DEVICE ERROR LOGGING ROUTINE AND
08C0 1612 : FROM THE DIAGNOSTIC BUFFER FILL ROUTINE.
08C0 1613 :
08C0 1614 : INPUTS:
08C0 1615 :
08C0 1616 : R0 = ADDRESS OF REGISTER SAVE BUFFER.
08C0 1617 : R4 = ADDRESS OF ADAPTER CONFIGURATION REGISTER.
08C0 1618 : R5 = DEVICE UNIT UCB ADDRESS.
08C0 1619 :
08C0 1620 : OUTPUTS:
08C0 1621 :
08C0 1622 : THE CONTROLLER AND DRIVE REGISTERS ARE SAVED IN THE SPECIFIED BUFFER.
08C0 1623 :
08C0 1624 :
08C0 1625 DR_REGDUMP:
08C0 1626 MOVL #<RM EC2+4+MBASL_BCR+4+8+4>/4,(R0)+ ;REGISTER DUMP ROUTINE
08C0 1627 MOVL MBASL_CSR(R4),(R0)+ ;INSERT NUMBER OF DEVICE REGS
08C0 1628 MOVL MBASL_CR(R4),(R0)+ ;SAVE CONFIGURATION REGISTER
08C0 1629 MOVL UCB$SL_DR_SR(R5),(R0)+ ;SAVE CONTROL REGISTER
08C0 1630 MOVL MBASL_VAR(R4),(R0)+ ;SAVE STATUS REGISTER
08C0 1631 MOVL MBASL_BCR(R4),(R0)+ ;SAVE VIRTUAL ADDRESS REGISTER
08C0 1632 EXTZV #9,#8,-8(R0),R1 ;SAVE BYTE COUNT REGISTER
08C0 1633 MOVL MBASL_MAP(R4)[R1],(R0)+ ;GET FINAL MAP REGISTER NUMBER
08C0 1634 CLRL (R0)+ ;SAVE FINAL MAP REGISTER CONTENTS
08C0 1635 DECL R1 ;ASSUME NO PREVIOUS MAP REGISTER
08C0 1636 BLSS 10$ ;CALCULATE PREVIOUS MAP REGISTER NUMBER
08C0 1637 MOVL MBASL_MAP(R4)[R1],-4(R0) ;IF LSS NONE
08C0 1638 10$: MOVZBL #<RM EC2+4>/4,R1 ;SAVE PREVIOUS MAP REGISTER CONTENTS
08C0 1639 MOVZBL UCB$B_SLAVE+1(R5),R2 ;SET NUMBER OF DRIVE REGISTERS TO SAVE
08C0 1640 MOVAL MBASL_ERB(R4)[R2],R2 ;GET DRIVE OFFSET CONSTANT
08C0 1641 20$: MOVL (R2)+,(R0)+ ;GET ADDRESS OF DRIVE REGISTERS
08C0 1642 SOBGTR R1,20$ ;SAVE DRIVE REGISTER
08C0 1643 MOVZBL UCB$B_DR_ERL(R5),(R0)+ ;ANY MORE TO SAVE?
08C0 1644 RSB ;SAVE MEDIUM OFFLINE INDICATOR
08C0 1645 :
```

51 F8 A0 08 09 EF 08D7 1632  
80 0800 C441 D0 08DD 1633  
80 04 A4 D0 08C6 1628  
80 00CC C5 D0 08CA 1629  
80 0C A4 D0 08CF 1630  
80 10 A4 D0 08D3 1631  
FC A0 0800 C441 D0 08E9 1637  
51 10 9A 08F0 1638  
52 0091 C5 9A 08F3 1639  
52 0400 C442 DE 08F8 1640  
80 82 D0 08FE 1641  
FA 51 F5 0901 1642  
80 00D5 C5 9A 0904 1643  
05 0909 1644  
090A 1645



```
090A 1647      .SBTTL  DISK DRIVE INITIALIZATION
090A 1648
090A 1649      DR_UNIT_INIT - DISK DRIVE UNIT INITIALIZATION
090A 1650
090A 1651      THIS ROUTINE IS CALLED AT SYSTEM INITIALIZATION AND AT POWER RECOVERY TO SET
090A 1652      DRIVE PARAMETERS AND TO WAIT FOR ONLINE DRIVES TO SPIN UP.
090A 1653
090A 1654      INPUTS:
090A 1655
090A 1656          R4 = ADDRESS OF MBA CONFIGURATION STATUS REGISTER.
090A 1657          R5 = DEVICE UNIT UCB ADDRESS.
090A 1658
090A 1659      OUTPUTS:
090A 1660
090A 1661          UNIT PARAMETERS ARE ESTABLISHED AND THE DRIVE IS SPUN UP IF IT WAS ONLINE.
090A 1662
090A 1663      SPECIAL NOTES:
090A 1664          This routine performs several special operations to support power
090A 1665          failure recovery in the RP07. To provide an understanding of these
090A 1666          operations, power failure recovery within in the RP07 is discussed
090A 1667          first. Then, the special actions taken by this routine are discussed.
090A 1668
090A 1669          The power up sequence in a RP07 drive is best described in terms of a
090A 1670          series of numbered states. The state numbers are shown in the LED
090A 1671          readout on the micro-processor control pannel, the section labeled
090A 1672          "PROGRAM CODE." The following lists these states and gives the
090A 1673          author's understanding what they mean.
090A 1674
090A 1675      STATE      COMMENTS
090A 1676      00,11,22,....,FF These states occur upon restoration of DC power.
090A 1677                      Presumably they are related to micro-processor
090A 1678                      initialization and testing. During these states,
090A 1679                      no MASSBUS interaction with the drive is possible.
090A 1680
090A 1681      00,01,02      These states occur after the above states during power
090A 1682                      failure recovery or after the START/STOP switch is
090A 1683                      moved from the STOP to the START position. These
090A 1684                      states also are related to micro-processor and disk
090A 1685                      system testing. The disk system is not spinning
090A 1686                      during these states. During these states, no
090A 1687                      MASSBUS interaction with the drive is possible.
090A 1688
090A 1689      03          During power failure recovery, this is the state in
090A 1690                      which multiple RP07 drives on a single system will
090A 1691                      synchronize their attempts to spin their disk media.
090A 1692                      Limited communication with the drive via the MASSBUS
090A 1693                      is enabled while the drive is in this state. The
090A 1694                      drive type register can be read, and the clear-drive
090A 1695                      command is accepted. The drive status register also
090A 1696                      can be read while the drive is in this state. For
090A 1697                      between 20 and 40 milliseconds after this state is
090A 1698                      entered, however, the drive status register contains
090A 1699                      garbage -- probably all bits except ATA and ERR set,
090A 1700                      a remnant of some internal test. After this initial
090A 1701                      period, the drive status register contains reasonable,
090A 1702                      valid information.
090A 1703
```



090A 1704 :  
090A 1705 :  
090A 1706 :  
090A 1707 :  
090A 1708 :  
090A 1709 :  
090A 1710 :  
090A 1711 :  
090A 1712 :  
090A 1713 :  
090A 1714 :  
090A 1715 :  
090A 1716 :  
090A 1717 :  
090A 1718 :  
090A 1719 :  
090A 1720 :  
090A 1721 :  
090A 1722 :  
090A 1723 :  
090A 1724 :  
090A 1725 :  
090A 1726 :  
090A 1727 :  
090A 1728 :  
090A 1729 :  
090A 1730 :  
090A 1731 :  
090A 1732 :  
090A 1733 :

04,05,06,07,08 These states occur while the disk medium is spinning  
upto speed. While in these states, no MASSBUS  
interaction with the drive is possible.

The following aspects of this routine relate specifically to dealing  
with power failure recovery as practiced by the RP07.

- o The sieze port operation, performed near the beginning of this  
routine, also has the effect of waiting for the RP07 drive to  
reach state 03. To allow both wait operations -- the sieze  
port function and the wait for RP07 to reach state 03 function  
-- to be combined, EXESPWRTIMCHK is used to time both  
functions. When this routine is called for reasons other than  
power failure recovery, it establishes a 20 millisecond wait  
interval for EXESPWRTIMCHK.
- o Once access to the RP07 has been established, this routine  
proceeds to determine the drive type, that register can be  
read and contains valid.
- o Before proceeding to test for medium-online, however, this  
routine waits for 50 milliseconds. This is intended to allow  
the drive status register to reach a valid state.
- o The medium-online test will wait for the drive to spin up.  
Because all drive registers show zero while MASSBUS access to  
the drive is disabled, it will correctly wait throughout  
states 03, 04, 05, 06, 07, and 08.

```

      53  54  A5  3C 090A 1734 DR_UNIT_INIT:
0090 C5  53  90 090A 1735 MOVZWL UCBSW_UNIT(R5),R3 ;DISK DRIVE UNIT INITIALIZATION
      53  20  90 090E 1736 MOVB R3,UCBSB_SLAVE(R5) ;GET DRIVE UNIT NUMBER
0091 C5  53  C4 0913 1737 MULL #<107>/4,R3 ;SET SLAVE UNIT NUMBER
      53  0400 C443 DE 0916 1738 MOVB R3,UCBSB_SLAVE+1(R5) ;CALCULATE DRIVE OFFSET CONSTANT
7E 00000000 GF D0 091B 1739 MOVAL MBASL_ERB(R4)[R3],R3 ;SET SLAVE OFFSET CONSTANT
      0E 12 0921 1740 MOVL G^EXESGL_PWRDONE, -(SP) ;GET ADDRESS OF DRIVE CONTROL REGISTER
      00000000 GF 16 0928 1741 BNEQ 105$ ;Save current powerfail limit time.
      50  02  C1 092A 1742 JSB G^EXESREAD TODR ;Non-zero value indicates powerfail.
      04  A3  D4 0930 1743 ADDL3 #2,R0,G^EXESGL_PWRDONE ;If not powerfail, construct our
      0F  17  78 0938 1744 105$: CLRL RM_DS(R3) ;limit time for port seizure.
52 04 A3 17 78 093B 1745 110$: ASHL #31-RM_DS_V_DPR, RM_DS(R3) ;Attempt to seize port.
      0F  19  19 0940 1746 120$: BLSS 120$ ;Did we seize the port?
      00000000 GF 16 0942 1747 JSB G^EXESPWRTIMCHK ;If LSS, we seized the port.
      F0  50  E8 0948 1748 BLBS R0, 110$ ;Wait for port to be siezed.
      64  A5  10 AA 094B 1749 BICW #UCBSM_ONLINE, UCBSW_STS(R5) ;Branch if haven't waited long enough.
      5D  11  11 094F 1750 BRB 15$ ;If never get the port,
52 18 A3 14 78 0951 1751 120$: ASHL #31-RM_DT_V_DRQ, RM_DT(R3) ;mark the drive offline and invalid.
      05  18  18 0956 1752 5$ BGEQ 5$ ;Is there a dualport kit?
      00D4 C5 08 88 0958 1753 BISB #DR_M_DUALPORT, - ;If GEQ, no dualport kit; continue.
      64  A5  10 A8 095D 1754 UCBSB_DR_STS(R5) ;Else, set flag indicating that disk
      00A5 30 0961 1755 5$: BISW #UCBSM_ONLINE,UCBSW_STS(R5) ;has a dualport kit.
45 64 A5 04 E1 0964 1756 BSBW DR_DTYPE ;SET UNIT ONLINE
4B 64 A5 08 E1 0969 1757 BBC #UCBSV_ONLINE,UCBSW_STS(R5), 15$ ;CLASSIFY DRIVE TYPE
41 A5 07 91 096E 1758 BBC #UCBSV_VALID,UCBSW_STS(R5), 30$ ;IF CLR, UNKNOWN DRIVE TYPE
      27 12 0972 1760 CMPB #DT$_RP07, UCBSB_DEVTYPE(R5) ;IF CLR, VOLUME SOFTWARE INVALID
      BNEQ 10$ ;Is this a RP07?
      ; Branch if not a RP07.
```



```
51 00000000'GF 50 01 3C 0974 1761 .SHOW MEB
                                0974 1762 TIMEWAIT -
                                0974 1763 time = #5000, - ; If this is a RP07,
                                0974 1764 bitval = #0, - ; wait for it to finish its
                                0974 1765 source = #0, - ; cup of coffee.
                                0974 1766 context = B
                                0974 1766 #SS$ NORMAL, R0
                                0977 MULL3 #5000, G^EXE$GL_TENUSEC, R1
                                0983 CLRL -(SP)
                                30010$: BITB #0, #0
                                BNEQ 30011$
                                0985 30010$: MOVL G^EXE$GL_UBDELAY, (SP)
                                0988 30012$: SOBGTR (SP), 30012$
                                0991 SOBGTR R1, 30010$
                                0994 CLRL R0
                                0997 30011$: TSTL (SP)+
                                0999 .NOSHOW MEB
                                0999 1767 10$: MOVZBL #F_DRVCLR!1, RM_CS1(R3) ; CLEAR DRIVE
                                099B 1768 10$: ASHL #3T-RM_DS_V_MOC, RM_DS(R3), R2 ; MEDIUM ONLINE?
                                099E 1769 10$: BLSS 20$ ; IF LSS YES
                                09A3 1770 10$: JSB G^EXE$PWRIMCHK ; CHECK FOR MAXIMUM TIME EXCEEDED
                                09A5 1771 10$: BLBS R0, 10$ ; IF LBS MORE TIME TO GO
                                09AB 1772 15$: BICW #UCB$M_VALID, UCB$W_STS(R5) ; MARK THE VOLUME INVALID
                                09AE 1773 15$: BRB 30$
                                09B4 1774 20$: MOVZBL #F_PACKACK!1, RM_CS1(R3) ; ACKNOWLEDGE PACK
                                09B6 1775 20$: MOVZBL #F_RELEASE!1, RM_CS1(R3) ; CLEAR DRIVE
                                09B9 1776 30$: BISL MBA$SR(R4), MBA$SR(R4) ; CLEAR MBA STATUS
                                09BC 1777 40$: TSTL (SP)+
                                09C1 1778 40$: BNEQ 50$ ; If powerfail limit time was zero
                                09C3 1779 40$: CLRL G^EXE$GL_PWRDONE ; when we started, make sure its
                                09C5 1780 40$: RSB ; zero when we leave.
                                09CB 1781 50$:
```

```
52 04 A3 13 78 099E 1769 10$:
    00000000'GF 11 19 09A3 1770 10$:
    ED 50 E8 09AB 1772 15$:
64 A5 0800 8F AA 09AE 1773 15$:
    03 11 09B4 1774 20$:
    63 13 9A 09B6 1775 20$:
    63 0B 9A 09B9 1776 30$:
    08 A4 08 A4 C8 09BC 1777 40$:
    8E D5 09C1 1778 40$:
    06 12 09C3 1779 40$:
    00000000'GF D4 09C5 1780 40$:
    05 09CB 1781 50$:
```



```

09CC 1783      .SBTTL  UNSOLICITED INTERRUPT ROUTINE
09CC 1784      :
09CC 1785      : DR_UNSolNT - UNSOLICITED INTERRUPT ROUTINE
09CC 1786      :
09CC 1787      : THIS ROUTINE IS CALLED WHEN AN UNSOLICITED ATTENTION CONDITION IS DETECTED.
09CC 1788      :
09CC 1789      : INPUTS:
09CC 1790      :
09CC 1791      :      R4 = ADDRESS OF CONFIGURATION STATUS REGISTER.
09CC 1792      :      R5 = DEVICE UNIT UCB ADDRESS.
09CC 1793      :
09CC 1794      : OUTPUTS:
09CC 1795      :
09CC 1796      :      IF VOLUME VALID IS CLEAR, THEN SOFTWARE VOLUME VALID IS CLEARED. THE
09CC 1797      :      UNIT STATUS IS CHANGED TO ONLINE AND THE DRIVE TYPE AND PARAMETERS ARE
09CC 1798      :      CLASSIFIED.
09CC 1799      :
09CC 1800      :
09CC 1801      : DR_UNSolNT:
09CC 1802      :      MOVZBL  UCBSB_SLAVE+1(R5),R3      ;UNSolICITED INTERRUPT
09CC 1803      :      MOVAL   MBASL_ERB(R4)[R3],R3      ;GET DRIVE OFFSET CONSTANT
09CC 1804      :      BISW    #UCBSM_ONLINE,UCBSW_STS(R5) ;SET UNIT ONLINE
09CC 1805      :      BSBW    DR_DTYPE      ;CLASSIFY DRIVE TYPE
09CC 1806      :      BBC     #UCBSV_ONLINE,UCBSW_STS(R5),10$ ;IF CLR, UNKNOWN DRIVE TYPE
09CC 1807      :      BBC     #UCBSV_VALID,UCBSW_STS(R5),20$ ;IF CLR, VOLUME SOFTWARE INVALID
09CC 1808      :      ASHL    #31-RM_DS_V_MOL,RM_DS(R3),R2 ;MEDIUM ONLINE?
09CC 1809      :      BGEQ    10$      ;IF GEQ NO
09CC 1810      :      BBC     #UCBSV_BSY,UCBSW_STS(R5),5$ ;We know the drive is online; thus,
09CC 1811      :      CMPB    #CDF_PACKACK,UCBSB_CEX(R5) ;if busy doing a PACKACK function,
09CC 1812      :      BEQL    20$      ;then don't clear software valid.
09CC 1813      :      ASHL    #31-RM_DS_V_VV,RM_DS(R3),R2 ;VOLUME VALID?
09CC 1814      :      BLSS    20$      ;IF LSS YES
09CC 1815      :      BICW    #UCBSM_VALID,UCBSW_STS(R5) ;CLEAR SOFTWARE VOLUME VALID
09CC 1816      :      RSB

```



```
0A09 1818 .SBTTL CLASSIFY DRIVE TYPE AND SET PARAMETERS
0A09 1819 :
0A09 1820 : RM_DTYPE - CLASSIFY DRIVE TYPE AND SET PARAMETERS
0A09 1821 :
0A09 1822 : THIS ROUTINE IS CALLED WHEN AN UNSOLICITED INTERRUPT OCCURS ON A DRIVE, DURING
0A09 1823 : SYSTEM INITIALIZATION, AND AT POWER RECOVERY TO DETERMINE THE DRIVE TYPE AND
0A09 1824 : SET UNIT PARAMETERS.
0A09 1825 :
0A09 1826 : INPUTS:
0A09 1827 :
0A09 1828 : R3 = ADDRESS OF DRIVE CONTROL REGISTER.
0A09 1829 : R4 = ADDRESS OF MBA CONFIGURATION STATUS REGISTER.
0A09 1830 : R5 = DEVICE UNIT UCB ADDRESS.
0A09 1831 :
0A09 1832 : OUTPUTS:
0A09 1833 :
0A09 1834 : THE DRIVE TYPE REGISTER IS INTERROGATED AND UNIT PARAMETERS ARE SET.
0A09 1835 :
0A09 1836 :
0A09 1837 DR_DTYPE:
0A09 1838 PUSHL RM_DT(R3) ;CLASSIFY DRIVE TYPE AND SET PARAMETERS
6E 18  A3 DD 0A09 1839 BICW #^C<^X1FF>,(SP) ;READ DRIVE TYPE REGISTER
52 FE00 8F AA 0A0C 1839 BICW #^C<^X1FF>,(SP) ;CLEAR EXTRANEOUS BITS
52 F627 CF 9E 0A11 1840 MOVAB DR_DTDESC,R2 ;GET ADDRESS OF DESCRIPTOR TABLE
82 6E B1 0A16 1841 10$: CMPW (SP),(R2)+ ;DRIVE TYPE MATCH?
OE 13 0A19 1842 BEQL 20$ ;IF EQL YES
52 OD C0 0A1B 1843 ADDL #DR_DTDESCLEN-2,R2 ;ADVANCE TO NEXT ENTRY
62 B5 0A1E 1844 TSTW (R2) ;END OF TABLE?
F4 12 0A20 1845 BNEQ 10$ ;IF NEQ NO
64 A5 10 AA 0A22 1846 BICW #UCBSM_ONLINE,UCBSW_STS(R5) ;SET UNIT OFFLINE
52 OD C2 0A26 1847 SUBL #DR_DTDESCLEN-2,R2 ;BACK UP TO LAST DRIVE DESCRIPTOR
41 A5 82 90 0A29 1848 20$: MOVW (R2)+,UCBSB_DEVTYPE(R5) ;SET DEVICE TYPE
44 A5 82 D0 0A2D 1849 MOVW (R2)+,UCBSL_DEVDEPEND(R5) ;SET DISK PACK GEOMETRY
00B0 C5 82 D0 0A31 1850 MOVW (R2)+,UCBSL_MAXBLOCK(R5) ;SET MAXIMUM BLOCKS PER PACK
008C C5 62 D0 0A36 1851 MOVW (R2),UCBSL_MEDIA_ID(R5) ;SET MEDIA IDENT
8E 05 D5 0A3B 1852 TSTL (SP)+ ;REMOVE DRIVE TYPE FROM STACK
0A3D 1853 RSB ;
0A3E 1854 DR_END: ;ADDRESS OF LAST LOCATION IN DRIVER
0A3E 1855
0A3E 1856 .END
```



DRDRIVER  
Symbol table

- RM03/RM05/RM80/RP07 DISK DRIVER H 3

15-SEP-1984 23:52:45 VAX/VMS Macro V04-00  
6-SEP-1984 21:02:04 [DRIVER.SRC]DRDRIVER.MAR;2

Page 36  
(1)

\$\$\$	= 00000020	R	02	DPTSM_SVP	= 00000002		
\$\$OP	= 00000002			DPT\$REINITAB	0000006A	R	02
ACPSACCESS	*****	X	03	DPT\$TAB	00000000	R	02
ACPSDEACCESS	*****	X	03	DR\$DDT	00000000	RG	03
ACPSMODIFY	*****	X	03	DRVCLR	00000254	R	03
ACPSMOUNT	*****	X	03	DR_DTDESC	0000003C	R	03
ACPSREADBLK	*****	X	03	DR_DTDESCLEN	= 0000000F		
ACPSWRITEBLK	*****	X	03	DR_DTYPE	00000A09	R	03
APPLY_ECC	0000038E	R	03	DR_END	00000A3E	R	03
ATS_MBA	= 00000000			DR_FUNCABLE	000000B2	R	03
AVAILABLE	00000245	R	03	DR_M_DCK	= 00000001		
CDF_AVAILABLE	= 00000011			DR_M_DUALPORT	= 00000008		
CDF_DIAGNOSE	= 00000014			DR_M_ECC_DEFER	= 00000010		
CDF_DRVCLR	= 00000004			DR_M_NOECC	= 00000004		
CDF_NOP	= 00000005			DR_M_OM	= 00000002		
CDF_OFFSET	= 00000006			DR_REGDUMP	000008C0	R	03
CDF_PACKACK	= 00000008			DR_STARTIO	00000146	R	03
CDF_READDATA	= 0000000C			DR_UNIT_INIT	0000090A	R	03
CDF_READHEAD	= 0000000E			DR_UNSOENT	000009CC	R	03
CDF_READPRESET	= 00000013			DR_V_DCK	= 00000000		
CDF_READTRACKD	= 00000010			DR_V_DUALPORT	= 00000003		
CDF_RECAL	= 00000003			DR_V_ECC_DEFER	= 00000004		
CDF_RETCENTER	= 00000007			DR_V_NOECC	= 00000002		
CDF_SEARCH	= 00000009			DR_V_OM	= 00000001		
CDF_SEARCHA	= 00000015			DTS_RM03	= 00000006		
CDF_SEEK	= 00000002			DTS_RM05	= 0000000F		
CDF_WRITECHECK	= 0000000A			DTS_RM80	= 0000000D		
CDF_WRITECHECKH	= 00000012			DTS_RP07	= 00000007		
CDF_WRITEDATA	= 0000000B			DYN\$C_DDB	= 00000006		
CDF_WRITEHEAD	= 0000000D			DYN\$C_DPT	= 0000001E		
CDF_WRITETRACKD	= 0000000F			DYN\$C_UCB	= 00000010		
CFUNC	00000785	R	03	ECC	00000331	R	03
CHECKRETRY	000002E7	R	03	EMBSL_DV_REGSAV	= 0000004E		
CHECKTAB	00000038	R	03	ENBXIT	000006C8	R	03
CHECKXT	00000300	R	03	ERL\$DEVICERR	*****	X	03
CRBSL_INTD	= 00000024			ERL\$DEVICTMO	*****	X	03
DATACHECK	0000029A	R	03	ERROR	000006FD	R	03
DC\$_DISK	= 00000001			EXESGL_PWRDONE	*****	X	03
DDB\$K_PACK	= 00000001			EXESGL_TENUSEC	*****	X	03
DDB\$K_ACPD	= 00000010			EXESGL_UBDELAY	*****	X	03
DDB\$K_DDT	= 0000000C			EXESIOFORK	*****	X	03
DEFER_ECC	000003B4	R	03	EXESLCLDSKVALID	*****	X	03
DEVSM_AVL	= 00040000			EXESONEPARM	*****	X	03
DEVSM_DIR	= 00000008			EXESPWRTIMCHK	*****	X	03
DEVSM_DUA	= 00008000			EXESREAD_TODR	*****	X	03
DEVSM_ELQ	= 00400000			EXESSENSEMODE	*****	X	03
DEVSM_FOD	= 00004000			EXESSETCHAR	*****	X	03
DEVSM_IDV	= 04000000			EXESZEROPARM	*****	X	03
DEVSM_NNM	= 00000200			EXFNC	00000628	R	03
DEVSM_ODV	= 08000000			FATALERR	00000473	R	03
DEVSM_RND	= 10000000			FDISPATCH	000001BA	R	03
DEVSM_SHR	= 00010000			FEX	00000556	R	03
DFUNC	000007AC	R	03	FTAB	00000098	R	03
DIAGNOSE	0000028A	R	03	FUNCAB_LEN	= 00000094		
DPT\$C_LENGTH	= 00000038			FUNCXT	0000050B	R	03
DPT\$C_VERSION	= 00000004			F_AVAILABLE	= 00000000		
DPT\$INITAB	00000038	R	02	F_DIAGNOSE	= 0000001C		



DRDRIVER  
Symbol table

- RM03/RM05/RM80/RP07 DISK DRIVER 1 3

15-SEP-1984 23:52:45 VAX/VMS Macro V04-00  
6-SEP-1984 21:02:04 [DRIVER.SRC]DRDRIVER.MAR;2

Page 37  
(1)

F_DRVCLR	= 00000008		
F_NOP	= 00000000		
F_OFFSET	= 0000000C		
F_PACKACK	= 00000012		
F_READDATA	= 00000038		
F_READHEAD	= 0000003A		
F_READPRESET	= 00000010		
F_READTRACKD	= 0000003C		
F_RECAL	= 00000006		
F_RELEASE	= 0000000A		
F_RETCENTER	= 0000000E		
F_SEARCH	= 00000018		
F_SEARCHA	= 00000018		
F_SEEK	= 00000004		
F_WRITECHECK	= 00000028		
F_WRITECHECKH	= 0000002A		
F_WRITEDATA	= 00000030		
F_WRITEHEAD	= 00000032		
F_WRITETRACKD	= 00000034		
GO	= 00000575	R	03
IDBSL_OWNER	= 00000004		
IMMED	= 000005A6	R	03
IOSV_COMMOD	= 00000006		
IOSV_DATACHECK	= 0000000E		
IOSV_DIAGNOSTIC	= 00000008		
IOSV_INHRETRY	= 0000000F		
IOSV_INHSEEK	= 0000000C		
IOSV_MOVETRACKD	= 00000007		
IOSV_SKPSECINH	= 00000009		
IOS_ACCESS	= 00000032		
IOS_ACPCONTROL	= 00000038		
IOS_AVAILABLE	= 00000011		
IOS_CREATE	= 00000033		
IOS_DEACCESS	= 00000034		
IOS_DELETE	= 00000035		
IOS_DIAGNOSE	= 0000001D		
IOS_DRVCLR	= 00000004		
IOS_MODIFY	= 00000036		
IOS_MOUNT	= 00000039		
IOS_NOP	= 00000000		
IOS_OFFSET	= 00000006		
IOS_PACKACK	= 00000008		
IOS_READHEAD	= 0000000E		
IOS_READBLK	= 00000021		
IOS_READPBLK	= 0000000C		
IOS_READPRESET	= 00000019		
IOS_READTRACKD	= 00000010		
IOS_READVBLK	= 00000031		
IOS_RECAL	= 00000003		
IOS_RELEASE	= 00000005		
IOS_RETCENTER	= 00000007		
IOS_SEARCH	= 00000009		
IOS_SEEK	= 00000002		
IOS_SENSECHAR	= 0000001B		
IOS_SENSEMODE	= 00000027		
IOS_SETCHAR	= 0000001A		
IOS_SETMODE	= 00000023		

IOS_UNLOAD	= 00000001		
IOS_VIRTUAL	= 0000003F		
IOS_WRITECHECK	= 0000000A		
IOS_WRITECHECKH	= 00000018		
IOS_WRITEHEAD	= 0000000D		
IOS_WritelBLK	= 00000020		
IOS_WRITEPBLK	= 0000000B		
IOS_WritETRACKD	= 0000000F		
IOS_WRITEVBLK	= 00000030		
IOCSAPPLYECC	*****	X	03
IOCSDIAGBUFILL	*****	X	03
IOCSLOADMBAMAP	*****	X	03
IOCSMNTVER	*****	X	03
IOCSRELCHAN	*****	X	03
IOCSREQCOM	*****	X	03
IOCSREQPCHANL	*****	X	03
IOCSRETURN	*****	X	03
IOCSUPDATRANSF	*****	X	03
IOCSWFIKPC	*****	X	03
IRPSL_MEDIA	= 00000038		
IRPSL_SWAPTE	= 0000002C		
IRPSS_FCODE	= 00000006		
IRPSV_FCODE	= 00000000		
IRPSV_FUNC	= 00000001		
IRPSV_PHYSIO	= 00000008		
IRPSW_BCNT	= 00000032		
IRPSW_FUNC	= 00000020		
IRPSW_STS	= 0000002A		
LDCYL	= 00000622	R	03
MASKH	= 00000008		
MASKL	= 04000000		
MBASL_BCR	= 00000010		
MBASL_CR	= 00000004		
MBASL_CSR	= 00000000		
MBASL_ERB	= 00000400		
MBASL_MAP	= 00000800		
MBASL_SR	= 00000008		
MBASL_VAR	= 0000000C		
MBASH_CR_ABORT	= 00000002		
MBASH_CR_IE	= 00000004		
MBASH_CR_INIT	= 00000001		
MBASH_ERROR	= 000E5FFF		
MBASH_SR_DLT	= 00000800		
MBASH_SR_ERCONF	= 00000008		
MBASH_SR_INVMAP	= 00000010		
MBASH_SR_ISTO	= 00000002		
MBASH_SR_MAPPE	= 00000020		
MBASH_SR_MBEXC	= 00000080		
MBASH_SR_MCPE	= 00020000		
MBASH_SR_MDPE	= 00000040		
MBASH_SR_MXF	= 00000100		
MBASH_SR_NED	= 00040000		
MBASH_SR_PGE	= 00080000		
MBASH_SR_RDS	= 00000004		
MBASH_SR_RDTO	= 00000001		
MBASH_SR_SPE	= 00004000		
MBASH_SR_WCKLWR	= 00000200		



DRDRIVER  
Symbol table

- RMC3/RM05/RM80/RP07 DISK DRIVER J 3

15-SEP-1984 23:52:45 VAX/VMS Macro V04-00  
6-SEP-1984 21:02:04 [DRIVER.SRC]DRDRIVER.MAR;2

Page 38  
(1)

```

MBASH_SR_WCKUPR      = 00000400
MBASV_SR_NED         = 00000012
NOP                  = 00000254 R 03
NORMAL               = 000002FD R R 03
OFF                  = 00000389 R R 03
OFFSET              = 00000254 R 03
OFFSIZ               = 00000004
OFFTAB               = 000000AE R R 03
PACKACK              = 0000024E R R 03
POSIT                = 0000061C R 03
PR$ IPL              = 00000012
READDATA             = 0000027C R 03
READHEAD             = 0000027C R R 03
READPRESET           = 00000254 R R 03
READTRACKD           = 00000261 R R 03
RECAL                = 00000254 R R 03
RELEASE              = 00000254 R R 03
RESETXFR             = 000008A1 R R 03
RETCENTER            = 00000254 R R 03
RETREG               = 000006CB R R 03
RETRY                = 0000032E R R 03
RETRYERR             = 00000441 R 03
RM_AS                = 00000010
RM_CS1               = 00000000
RM_CS1_M_GO          = 00000001
RM_DA                = 00000014
RM_DC                = 00000028
RM_DS                = 00000004
RM_DS_M_DPR          = 00000100
RM_DS_M_ERR          = 00004000
RM_DS_V_DPR          = 00000008
RM_DS_V_ERR          = 0000000E
RM_DS_V_MOL          = 0000000C
RM_DS_V_VV           = 00000006
RM_DT                = 00000018
RM_DT_V_DRQ          = 0000000B
RM_ECT               = 00000038
RM_EC2               = 0000003C
RM_ER1               = 00000008
RM_ER1_M_AOE         = 00000200
RM_ER1_M_DCK         = 00008000
RM_ER1_M_DTE         = 00001000
RM_ER1_M_ECH         = 00000040
RM_ER1_M_FER         = 00000010
RM_ER1_M_HCE         = 00000080
RM_ER1_M_HCRC        = 00000100
RM_ER1_M_IAE         = 00000400
RM_ER1_M_ILF         = 00000001
RM_ER1_M_ILR         = 00000002
RM_ER1_M_OPI         = 00002000
RM_ER1_M_PAR         = 00000008
RM_ER1_M_RMR         = 00000004
RM_ER1_M_UN$         = 00004000
RM_ER1_M_WCF         = 00000020
RM_ER1_M_WLE         = 00000800
RM_ER1_V_FER         = 00000004
RM_ER1_V_HCE         = 00000007

```

```

RM_ER1_V_HCRC        = 00000008
RM_ER1_V_OPI         = 0000000D
RM_ER1_V_UN$         = 0000000E
RM_ER1_V_WLE         = 0000000B
RM_ER2               = 00000034
RM_ER2_M_BSE         = 00008000
RM_ER2_M_DPE         = 00000008
RM_ER2_M_DVC         = 00000080
RM_ER2_M_IVC         = 00001000
RM_ER2_M_LBC         = 00000400
RM_ER2_M_LSC         = 00000800
RM_ER2_M_OPE         = 00002000
RM_ER2_V_BSE         = 0000000F
RM_ER2_V_SKI         = 0000000E
RM_ER2_V_SSE         = 00000005
RM_LA                = 0000001C
RM_MR                = 0000000C
RM_MR2               = 00000030
RM_MR_M_DM           = 00008000
RM_OF                = 00000024
RM_OF_M_CMO          = 00008000
RM_OF_M_FMT          = 00001000
RM_OF_M_HCI          = 00000400
RM_OF_M_MTD          = 00004000
RM_OF_M_SSEI         = 00000200
RM_OF_V_SSEI         = 00000009
RM_SN                = 00000020
RM_UNUSED             = 0000002C
SAFUNC               = 00000777 R 03
SEARCH               = 00000254 R R 03
SEARCHA              = 000005F9 R R 03
SEEK                 = 00000254 R R 03
SEIZE                = 000005BD R 03
SIZ...               = 00000001
SPECOND              = 00000848 R 03
SS$_CTRLERR          = 00000054
SS$_DATACHECK        = 0000005C
SS$_DRVERR           = 0000008C
SS$_FORMAT           = 000000BC
SS$_IVADDR           = 00000134
SS$_MEDOFL           = 000001A4
SS$_NONEXDRV         = 000001C4
SS$_NORMAL           = 00000001
SS$_OPINCOMPL        = 000002D4
SS$_PARITY           = 000001F4
SS$_TIMEOUT          = 0000022C
SS$_UNSAFE           = 0000023C
SS$_VOLINV           = 00000254
SS$_WASECC           = 00000639
SS$_WRITLCK          = 0000025C
TRANNOCH             = 00000290 R R 03
TRANRQCH             = 0000028A R R 03
TRANXT               = 00000303 R 03
UCB$B_CEX            = 00000093
UCB$B_DEVCLASS       = 00000040
UCB$B_DEVTYPE        = 00000041
UCB$B_DIPL           = 0000005E

```



DRDRIVER  
Symbol table

- RM03/RM05/RM80/RP07 DISK DRIVER<sup>K 3</sup>

15-SEP-1984 23:52:45  
6-SEP-1984 21:02:04

VAX/VMS Macro V04-00  
[DRIVER.SRC]DRDRIVER.MAR;2

Page 39  
(1)

UCBSB_DR_ERL	000000D5		
UCBSB_DR_SSTS	000000D4		
UCBSB_ERTCNT	= 00000080		
UCBSB_ERTMAX	= 00000081		
UCBSB_FEX	= 00000092		
UCBSB_FIPL	= 0000000B		
UCBSB_OFFNDX	= 000000CA		
UCBSB_OFFRTC	= 000000CB		
UCBSB_SECTORS	= 00000044		
UCBSB_SLAVE	= 00000090		
UCBSK_DR_LENGTH	= 000000DC		
UCBSK_LCC_DISK_LENGTH	= 000000CC		
UCBSL_CRB	= 00000024		
UCBSL_DEVCHAR	= 00000038		
UCBSL_DEVCHAR2	= 0000003C		
UCBSL_DEVDEPEND	= 00000044		
UCBSL_DPC	= 0000009C		
UCBSL_DR_BCR	000000D8		
UCBSL_DR_SR	000000CC		
UCBSL_IRP	= 00000058		
UCBSL_MAXBLOCK	= 000000B0		
UCBSL_MEDIA_ID	= 0000008C		
UCBSL_SVAPTE	= 00000078		
UCBSM_ONLINE	= 00000010		
UCBSM_POWER	= 00000020		
UCBSM_TIMEOUT	= 00000040		
UCBSM_VALID	= 00000800		
UCBSV_BSY	= 00000008		
UCBSV_ECC	= 00000000		
UCBSV_ONLINE	= 00000004		
UCBSV_POWER	= 00000005		
UCBSV_VALID	= 0000000B		
UCBSW_BCNT	= 0000007E		
UCBSW_DA	= 000000BC		
UCBSW_DC	= 000000BE		
UCBSW_DEVBUSIZ	= 00000042		
UCBSW_DEVSTS	= 00000068		
UCBSW_DR_ER2	000000D0		
UCBSW_DR_MR	000000D2		
UCBSW_DR_OFR	000000D6		
UCBSW_ECT	= 000000C4		
UCBSW_EC2	= 000000C6		
UCBSW_FUNC	= 0000009A		
UCBSW_OFFSET	= 000000C8		
UCBSW_STS	= 00000064		
UCBSW_UNIT	= 00000054		
UNLOAD	00000245	R	03
VECSL_IDB	= 00000008		
WRITECHECK	00000271	R	03
WRITECHECKH	00000271	R R	03
WRITEDATA	00000277	R R	03
WRITEHEAD	00000277	R R	03
WRITETRACKD	0000025C	R	03
XFER	0000060D	R	03



+-----+  
! Psect synopsis !  
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 ( 0.)	00 ( 0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABS\$	0000000C ( 220.)	01 ( 1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
\$\$\$105_PROLOGUE	00000070 ( 112.)	02 ( 2.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
\$\$\$115_DRIVER	00000A3E ( 2622.)	03 ( 3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC LONG

+-----+  
! Performance indicators !  
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	31	00:00:00.04	00:00:03.11
Command processing	109	00:00:00.38	00:00:05.85
Pass 1	605	00:00:19.66	00:01:54.25
Symbol table sort	0	00:00:02.56	00:00:22.81
Pass 2	331	00:00:04.67	00:00:31.08
Symbol table output	47	00:00:00.23	00:00:00.56
Psect synopsis output	2	00:00:00.01	00:00:00.25
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	1127	00:00:27.57	00:02:57.92

The working set limit was 2250 pages.

159994 bytes (313 pages) of virtual memory were used to buffer the intermediate code.

There were 130 pages of symbol table space allocated to hold 2372 non-local and 84 local symbols.

1856 source lines were read in Pass 1, producing 24 object records in Pass 2.

48 pages of virtual memory were used to define 45 macros.

+-----+  
! Macro library statistics !  
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	30
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	10
TOTALS (all libraries)	40

2486 GETS were required to define 40 macros.

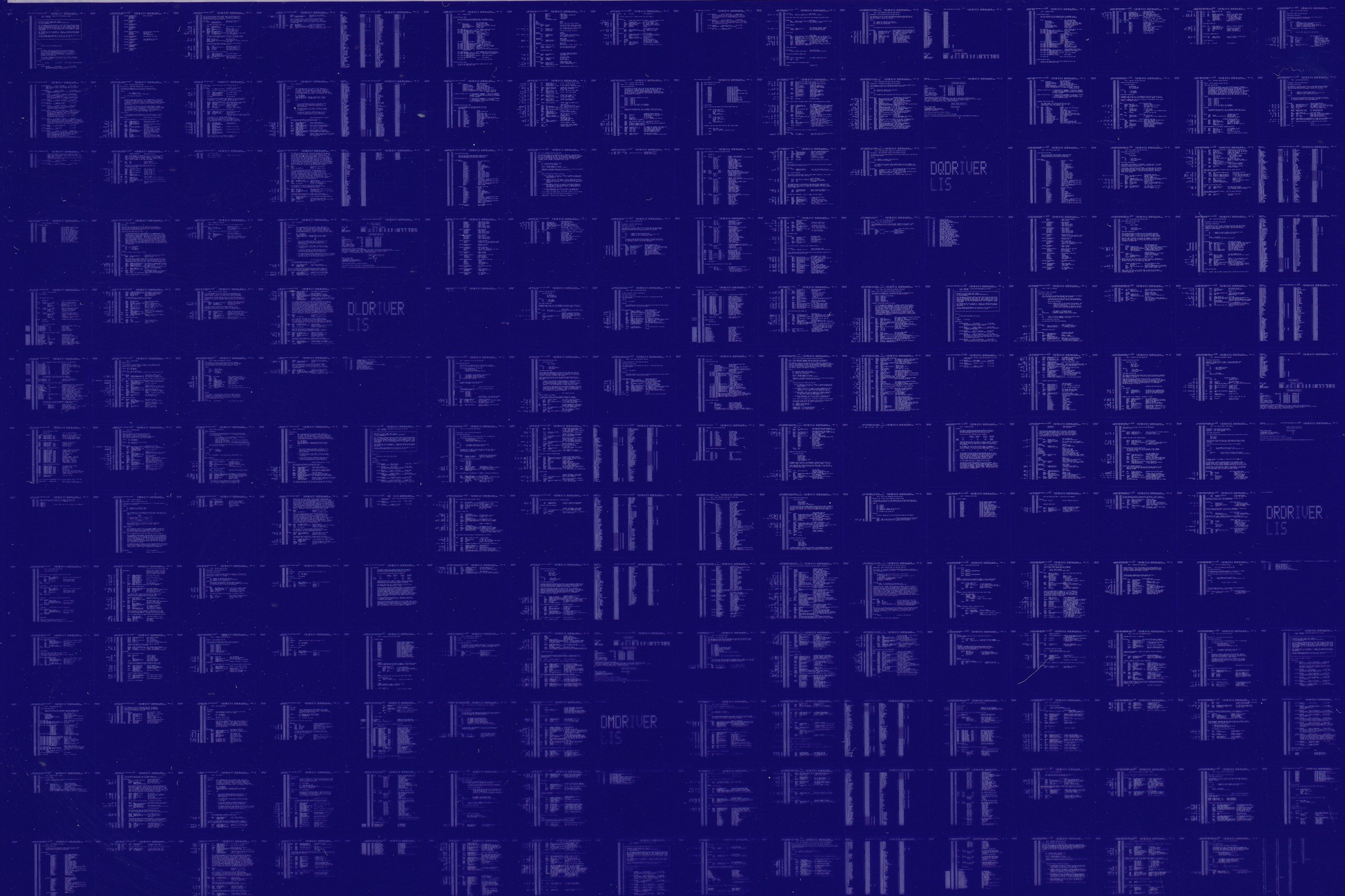
There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:DRDRIVER/OBJ=OBJ\$:DRDRIVER MSRC\$:DRDRIVER/UPDATE=(ENH\$:DRDRIVER)+EXECMLS/LIB



0109 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY





0110 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

